



TEKNILLINEN KORKEAKOULU

Sähkö- ja tietoliikennetekniikan osasto

Teemu Hyle

LANGATON RAPORTOINTIJÄRJESTELMÄ

Diplomityö, joka on jätetty opinnäytteenä tarkastettavaksi diplomi-insinöörin tutkintoa varten Espoossa 28.2.2005

Työn valvoja

Professori Pirkko Oittinen

Työn ohjaaja

Diplomi-insinööri Markku Myllylä

Tekijä: Teemu Hyle

Työn nimi: Langaton raportointijärjestelmä

Päivämäärä: 28.2.2005

Sivumäärä: 65

Osasto: Sähkö- ja tietoliikennetekniikka

Professori: AS-75 Viestintätekniikka

Työn valvoja: Professori Pirkko Oittinen

Työn ohjaaja: DI Markku Myllylä

Tämän diplomityön tarkoitus on määritellä rahapeliautomaattien langattoman raportointijärjestelmän arkkitehtuuri. Tämä sisältää mm. käytettävien tiedonsiirtoprotokollien, kommunikointitapojen, ohjelma- ja tietokantarajapintojen, palvelinohjelmiston tarkastelua sekä relaatiotietokannan kuvauksen.

Tässä työssä tarkastellaan rahapeliautomaattien liiketoimintaympäristöä sekä menetelmiä tapahtumapohjaisissa järjestelmissä. Tapahtumapohjaisiin järjestelmiin kuuluvat kommunikaatiojärjestelmät sekä valvontajärjestelmät. Lisäksi käsitellään tietoliikennettä GSM-järjestelmässä, tietokantajärjestelmiä, rajapintoja eri järjestelmien välillä, eri palvelimia sekä olemassa olevia M2M-laitteita.

Avainsanat: langaton raportointi, tapahtumapohjainen järjestelmä, rahapeliautomaatti

Author: Teemu Hyle

Name of the Thesis: Wireless reporting system

Date: 28.2.2005

Number of pages: 65

Department of Electrical and Communications Engineering

Professorship: AS-75 Media Technology

Supervisor: Professor Pirkko Oittinen

Instructor: Markku Myllylä, M.Sc.

The aim of this master's thesis was to plan an architecture of a wireless reporting system for gaming machines. The specification includes data transfer protocols, process communications, application programming interfaces, database interfaces, server software and database definition.

The theoretical part of the thesis deals with the business environment of the slot machines and distributed event-based systems. Publish/subscribe communication and the monitoring and management of systems and networks are examples of event-based systems. Data communication in the GSM network, database management systems, interfaces, servers and existing M2M-systems are also covered in this thesis.

Keywords: wireless reporting, event based system, slot machine

ALKULAUSE

Kiitos kaikille työtovereille, jotka ovat olleet mukana tähän työhön liittyvissä projekteissa. Erityiset kiitokset esitän työn ohjaajalle DI Markku Myllylälle.

Espoossa 28.2.2005



Teemu Hyle

SISÄLLYSLUETTELO

1	JOHDANTO.....	1
1.1	TAUSTAA	1
1.2	DIPLOMITYÖN TAVOITE	3
2	LIIKETOIMINTAYMPÄRISTÖ	4
2.1	YLEISTÄ.....	4
2.2	TOIMINTAYMPÄRISTÖN OSAT.....	5
2.2.1	<i>Huoltomies</i>	5
2.2.2	<i>Rahankerääjä</i>	5
2.2.3	<i>Käyttäjä</i>	5
2.2.4	<i>Päämies</i>	6
2.2.5	<i>Operaattori</i>	6
2.2.6	<i>Automaattilaitteisto</i>	6
3	LIIKETOIMINNAN VAATIMUKSET.....	10
3.1	YLEISET VAATIMUKSET	10
3.1.1	<i>Yleistä</i>	10
3.1.2	<i>Reaaliaikaiset hälytykset</i>	10
3.1.3	<i>Säännöllinen tiedonkeruu</i>	10
3.1.4	<i>Joustava ympäristö</i>	10
3.1.5	<i>Kaksisuuntainen tietoliikenne</i>	11
3.1.6	<i>Luotettavuus</i>	11
3.1.7	<i>Langattomat päätelaitteet</i>	11
3.2	HENKILÖKUNTA JA KÄYTTÄJÄT	12
3.2.1	<i>Yleistä</i>	12
3.2.2	<i>Huoltomies</i>	12
3.2.3	<i>Rahankerääjä</i>	12
3.2.4	<i>Käyttäjä</i>	12
3.2.5	<i>Päämies</i>	12
3.2.6	<i>Operaattori</i>	12
3.3	AUTOMAATTILAITTEISTO.....	13
3.4	PALVELINYMPÄRISTÖ	13
4	TAPAHTUMAPOHJAISET JÄRJESTELMÄT.....	14
4.1	YLEINEN TOIMINTAPERIAATE	14
4.1.1	<i>Yleistä</i>	14
4.1.2	<i>Observer-malli</i>	14
4.2	KOMMUNIKAATIOJÄRJESTELMÄT.....	15
4.2.1	<i>Yleistä</i>	15
4.2.2	<i>CORBA</i>	15
4.2.3	<i>Olemassa olevia järjestelmiä</i>	16
4.3	VALVONTAJÄRJESTELMÄT	16
4.3.1	<i>Yleistä</i>	16
4.3.2	<i>Olemassa olevia järjestelmiä</i>	16
5	TIETOLIIKENNE GSM-JÄRJESTELMÄSSÄ.....	17
5.1	YLEISTÄ.....	17

5.2	PÄÄTELAITE.....	17
5.3	LYHYTSANOMAT.....	17
5.4	TIEDONSIIRTO	19
6	TIETOKANTAJÄRJESTELMÄ	20
6.1	YLEISTÄ.....	20
6.2	RELAATITIEDOKANNAT.....	20
6.3	TIETOKANNAN HALLINTAJÄRJESTELMÄT	21
6.3.1	<i>Yleistä</i>	21
6.3.2	<i>Oracle Database</i>	21
6.3.3	<i>SQL Server</i>	22
6.3.4	<i>MySQL</i>	23
6.3.5	<i>Tietotyypit</i>	24
6.4	SQL-KIELI	24
6.5	RAJAPINNAT	25
6.5.1	<i>Järjestelmäkohtaiset rajapinnat</i>	25
6.5.2	<i>ODBC</i>	25
6.5.3	<i>JDBC</i>	26
6.6	TIETOKANNAN LAUKAISIMET.....	27
7	PALVELIMET.....	29
7.1	YLEISTÄ.....	29
7.2	SMS-YHDYSKÄYTÄVÄ.....	29
7.3	WWW-PALVELIN	29
7.3.1	<i>Yleistä</i>	29
7.3.2	<i>CGI</i>	29
7.3.3	<i>PHP</i>	30
7.3.4	<i>HTML</i>	30
7.3.5	<i>WML</i>	31
7.3.6	<i>CSS</i>	31
7.4	SOVELLUSPALVELIMET	31
7.4.1	<i>Yleistä</i>	31
7.4.2	<i>J2EE</i>	31
7.5	RAJAPINNAT	32
7.5.1	<i>Yleistä</i>	32
7.5.2	<i>RPC-etäproseduurikutsu</i>	32
7.5.3	<i>HTTP</i>	34
7.5.4	<i>OTA</i>	35
7.6	SALAUSSALGORITMIT	35
7.6.1	<i>HTTPS</i>	35
7.6.2	<i>MD5</i>	36
8	M2M-JÄRJESTELMIÄ JA -LAITTEITA	37
8.1	YLEISTÄ.....	37
8.2	NOKIA M2M PLATFORM	37
8.2.1	<i>Yleistä</i>	37
8.2.2	<i>Sovelluskehitysalusta</i>	37
8.2.3	<i>Päätelaitteet</i>	38
8.3	SIEMENS M2M.....	38
9	JÄRJESTELMÄN MÄÄRITELMÄ.....	39

9.1	YLEISTÄ.....	39
9.2	JÄRJESTELMÄN PERUSRAKENNE.....	40
9.2.1	Viestien vastaanotto SMS-yhdyskäytävästä	40
9.2.2	Tietojen tallennus relaatiotietokantaan	41
9.2.3	Viestien käsittely	43
9.2.4	Hälytykset.....	44
9.2.5	Raportointi	45
9.3	TIETOKANTA.....	46
9.3.1	Yleistä.....	46
9.3.2	Automaatin tyyppi.....	46
9.3.3	Tyypikohtaiset tapahtumat	47
9.3.4	Tyypikohtainen tilastotieto.....	47
9.3.5	Automaatti.....	47
9.3.6	Automaatin tapahtumat.....	48
9.3.7	Automaatin tilastotieto.....	48
9.3.8	Saapuneet viestit	49
9.3.9	Lähtevät viestit	49
9.3.10	Järjestelmän käyttäjät.....	49
9.3.11	Hälytykset.....	50
9.3.12	Tietokantamalli	50
9.4	JÄRJESTELMÄN KÄYTTÖ JA YLLÄPITO	52
9.4.1	Yleinen käyttö.....	52
9.4.2	Ylläpito.....	52
9.4.3	Käyttö WAP-selaimella.....	52
9.5	RAPORTIT.....	53
9.5.1	Yleistä.....	53
9.5.2	Tilastoraportit	54
9.5.3	Tapahtumaraportit	54
9.5.4	Hälytysraportit	55
10	PILOTTIPROJEKTI	56
10.1	YLEISTÄ.....	56
10.2	JÄRJESTELMÄN KOMPONENTIT	56
10.2.1	Pelikoneet.....	56
10.2.2	Raportointiyksiköt	56
10.2.3	Palvelimet	57
10.2.4	Lyhytsanomapalvelut	57
10.2.5	Sovellukset.....	58
10.2.6	Käyttöliittymä ja raportit	58
10.3	PILOTTIPROJEKTIN JOHTOPÄÄTÖKSET	59
11	JOHTOPÄÄTÖKSET.....	60
11.1	YHTEENVETO	60
11.2	TAVOITTEIDEN TÄYTTYMINEN	60
11.2.1	Yleistä.....	60
11.2.2	Reaaliaikaiset hälytykset.....	60
11.2.3	Säännöllinen tiedonkeruu	61
11.2.4	Joustava ympäristö	61
11.2.5	Kaksisuuntainen tietoliikenne	61
11.2.6	Luotettavuus.....	61

11.2.7	<i>Langattomat päätelaitteet</i>	61
12	LÄHDELUETTELO	62

LYHENNELUETTELO

ADK	Application Development Kit
ANSI	American National Standards Institute
API	Application Program Interface
AWP	Amusement with Prize
BACTA	British Amusement Catering Trades Association
CGI	Common Gateway Interface
CSS	Cascading Style Sheets
CORBA	Common Object Request Broker Architecture
DBMS	Database Management System
ECA	Event-Condition-Action
EJB	Enterprise JavaBean
EPROM	Erasable Programmable Read Only Memory
ER	Entity Relationship
ETSI	European Telecommunications Standards Institute
GPRS	General Packet Radio Service
GPL	General Public License
GSM	Global System for Mobile Communications
HSCSD	High Speed Circuit Switched Data
HTML	Hypertext Markup Language
HTTP	Hyper Text Transfer Protocol
HTTPS	Hypertext Transfer Protocol over Secure Sockets Layer
IDL	Interface Definition Language
IIOP	Internet Inter-ORB Protocol
IMAP	Internet Message Access Protocol
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
ISO	International Organization for Standardization
JDBC	Java Database Connectivity
JMS	Java Message Service
JNDI	Java Naming and Directory Interface
J2EE	Java 2 Platform, Enterprise Edition
LDAP	Lightweight Directory Access Protocol
MSISDN	Mobile Subscriber International ISDN Number
M2M	Machine-To-Machine
OCI	Oracle Call Interface
OC CI	Oracle C++ Call Interface
ODBC	Open Database Connectivity
OTA	Over The Air
PDU	Protocol Data Unit
PHP	PHP: Hypertext Preprocessor

PL/SQL	Procedural Language/Structured Query Language
POP	Post Office Protocol
QoS	Quality of Service
RAM	Random Access Memory
RPC	Remote Procedure Call
SGML	Standard Generalized Markup Language
SM MO	Short Message Mobile Originated
SM MT	Short Message Mobile Terminated
SMS	Short Message Service
SMSC	Short Message Service Center,
SQL	Structured Query Language
SSL	Secure Sockets Layer
TCP/IP	Transmission Control Protocol/Internet Protocol
TSL	Transport Secure Layer
UML	Unified Modeling Language
USC	Universal Character Set
USSD	Unstructured Supplementary Service Data
WAP	Wireless Access Protocol
WML	Wireless Markup Language
WWW	World Wide Web
XHTML	Extensible HyperText Markup Language
XML	Extensible Markup Language

1 JOHDANTO

1.1 Taustaa

Lähes kaikki liiketoiminta yrittää tehostaa toimintaprosessejaan sekä saamaan tätä kautta kustannussäästöjä. Liiketoiminta raha-automaateilla ei tee tähän poikkeusta.

Raha-automaattien toimintaympäristön käyttötapauksiin kuuluvat mm. automaatin asennus toimipisteeseen, automaatin käyttö asiakkaan toimesta, automaatin kunnossapito, automaatin korjaus sekä automaatin ylläpito. Automaatin kunnossapito sisältää yleiset määräaikaistoimenpiteet kuten huollon. Kun automaatti vikaantuu, tapahtuu laitteen korjaustoimenpiteet. Raha-automaattien ylläpitoon kuuluvat rahojen lisäys ja keräys sekä tuotteiden lisäykset.

Rahalla toimivat automaatit voidaan jakaa kahteen ryhmään: rahapeliautomaatit ja myyntiautomaatit. Molemmat automaattityypit ovat samankaltaisia, sillä niissä ostetaan rahalla palveluita. Peliautomaatissa palvelu on itse peli, kun myyntiautomaatissa se on jokin fyysinen tuote. Molemmissa automaattityypeissä on myös rahanpalautustoiminto. Myyntiautomaatissa palautetaan mahdolliset vaihtorahat, kun taas peliautomaatista maksetaan mahdolliset voitot.

Raha-automaatit sisältävät sisäisiä laskureita, jotka pitävät tietoa laitteen yleisistä käyttötapahtumista. Tietty tapahtuma lisää aina laskurin arvoa. Laskureissa voi olla myös automaatin tilatietoja. Tällaisia tapahtumia ja tietoja ovat mm. seuraavat.

- Raha sisään
- Raha ulos
- Mahdollisten tuotteiden lukumäärä

Raha-automaatit sisältävät myös tapahtumatietoa, jotka aiheuttavat merkinnän kirjausketjuun myöhempää tarkastelua varten. Tällaisia ovat mm. seuraavat.

- Huolto-oven avaus
- Huolto-oven sulkeminen
- Rahalaatikon avaus

- Rahalaatikon sulkeminen

Laitteen omistaja tai operaattori tekee tarkastuskäyntejä automaateille. Käynneillä kerätään laskureiden tilatiedot sekä automaatin tapahtumatiedot. Tarkastuskäynnit voivat olla yhdessä huollon tai korjauksen kanssa.

Automaatin sijoituspaikan henkilökunta seuraa laitteiden toimintaa ja ilmoittaa mahdollisista vikatilanteista omistajalle, operaattorille tai määrättyyn huoltoyhtiöön. Henkilökunta voi suorittaa myös rahankeräyksen automaateista, jos siihen on annettu valtuutus. Henkilökunta raportoi rahamääristä suoraan operaattorille tai omistajalle.

Jotta tätä toimintaa voidaan tehostaa, täytyy liiketoiminnan tapahtumaketjuun tulla muutoksia. Tiedonkeruuta, rahankeruuta ja huoltotoimenpiteitä täytyy tehostaa. Laitteiston automaattinen raportointijärjestelmä on ratkaisu tähän.

Raportoinnilla voidaan saada tietoa hälytyksistä, joita kirjausketjuun saapuneet merkinnät ovat aiheuttaneet. Lisäksi voidaan kerätä yleistä tilastotietoa, jota saadaan laskureista. Kun tämä tapahtuu automaattisesti, voidaan automaatilla käyntiä vähentää. Raportoinnin avulla voidaan myös helpommin valtuuttaa kolmas osapuoli, kuten automaatin sijaintipaikan henkilökunta, keräämään rahat automaatista. Tämä on mahdollista, koska voidaan tarkistaa vastaavako automaatin lähettämät tiedot ja rahankerääjän ilmoittama tieto rahamäärästä. Tieto automaatin toimintakunnosta saadaan vikaraportoinnista tai raportoinnin puuttumisesta, joten vikatilanteisiin pystytään reagoimaan nopeammin. Säädetävissä oleva raportointiväli mahdollistaa esimerkiksi päiväkohtaisen myynnin seurannan heti päivän jälkeen. Raportointijärjestelmä voi mahdollistaa myös tarkempien tietojen kyselyt automaatille. Raportointijärjestelmä voi mahdollistaa myös sen, että huoltomiehille ja yritysjohdolle luodaan erilaiset raportit tarpeen mukaan.

Langattomuus raportointijärjestelmässä mahdollistaa raportoinnin alueilla, joissa ei ole kiinteää verkkoa. Tämä edesauttaa automaattien sijoittamista haja-asutusalueille. Langattomalla raportointijärjestelmällä varustetut automaatit ovat nopeampia asentaa toimipaikkaansa, koska ei tarvita liityntää kiinteään tietoliikenneverkkoon, joka voi olla kallista asentaa.

1.2 Diplomityön tavoite

Diplomityön tarkoitus on rahapeliautomaattien langattoman raportointijärjestelmän palvelinpuolen arkkitehtuurin määrittelemine. Tämä sisältää mm. järjestelmän yleisarkkitehtuurin, tiedonsiirtoprotokollat, ohjelmarajapinnat, palvelinohjelmistot ja raportoinnisen loppukäyttäjälle. Määrittelyssä otetaan huomioon liiketoiminnan asettamat vaatimukset, joita ovat mm. reaaliaikaiset hälytykset, säännöllinen tiedonkeruu sekä järjestelmän joustavuus mahdollisille laajennuksille.






2 LIIKETOIMINTAYMPÄRISTÖ

2.1 Yleistä

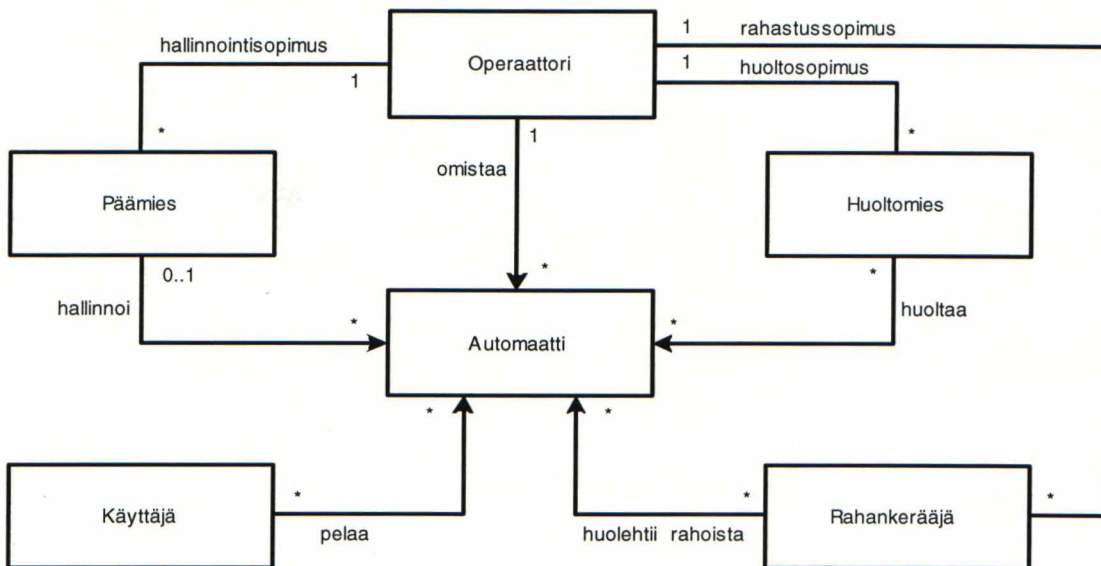
Lähtökohtana raportointijärjestelmän rakentamiseen on olemassa oleva rahapeliautomaattien toimintaympäristö. Ympäristöön voidaan katsoa kuuluvan seuraavia asioita.

- Huoltomies
- Rahankerääjä
- Käyttäjä
- Päämies
- Operaattori
- Automaattilaitteisto

Kuvassa 1 on esitetty liiketoimintaympäristön tekijöiden perussuhteet toisiinsa UML-kaaviona. Taulukossa 1 on kuvattu käytettävät merkinnät.

Merkintä	Kuvaus
	Assosiaatio eli kahden luokan välinen yhteys
	Lukumääräsuhde tasan yksi
	Lukumääräsuhde nolla tai yksi
	Lukumääräsuhde nolla tai enemmän
	Navigoitavuus luokkien välillä

Taulukko 1. UML-mallin merkintöjä /21/



Kuva 1. Liiketoimintaympäristön perussuhteet

2.2 Toimintaympäristön osat

2.2.1 Huoltomies

Huoltomiehen tehtävänä on huoltaa automaattia sekä tehdä korjauksia vikatilanteissa. Huoltotehtävät ovat määräaikaishuoltoja, joihin kuuluu mm. kuluvien osien vaihtaminen. Korjaustehtävät edellyttävät toimeksiantoa koneen päämieheltä, eli paikasta minne laite on sijoitettu. Vikatilanteet saatetaan kuitenkin huomata vasta määräaikaishuollon yhteydessä.

2.2.2 Rahankerääjä

Rahankerääjän tehtävänä on kerätä automaattiin kertyneet rahat ja raportoida tästä automaatin operaattorille. Huoltomies voi tehdä rahankeruun, mutta se voidaan valtuuttaa myös kolmannelle osapuolelle. Tällainen osapuoli on yleensä automaatin päämies eli automaatin sijoituspaikassa toimiva henkilö. Rahankerääjä lisää tarvittaessa myös rahaa maksukassaan, josta maksetaan voitot.

2.2.3 Käyttäjä

Automaatin käyttäjä on peliautomaatin pelaaja eli asiakas. Käyttäjä on automaatin tuotteen eli pelin ostaja.

2.2.4 Päämies

Päämies on toimipaikka, jonne automaatti on sijoitettu. Päämiehellä tarkoitetaan myös toimipaikassa toimivaa henkilöä. Päämiehen tehtäviin kuuluu valvoa automaatin yleistä kuntoa ja raportoida mahdollisista vikatilanteista operaattorille. Päämiehen tehtäviin voi kuulua myös rahankeruu automaatista.

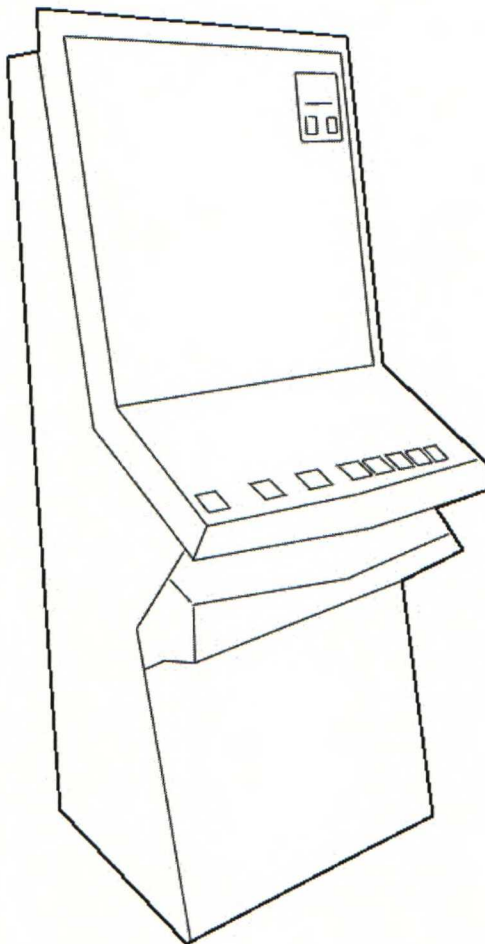
2.2.5 Operaattori

Operaattori on automaatin omistaja. Operaattori voi olla myös vuokrannut automaattilaitteistot todelliselta omistajalta, mutta tämän liiketoimintaympäristön ja automaatin kannalta operaattori on sen omistaja.

Operaattori pitää kirjaa automaattien sijainneista, saa automaattien keräämät rahat sekä valvoo niiden yleistä toimintaa.

2.2.6 Automaattilaitteisto

Rahapeliautomaatti on tämän liiketoimintajärjestelmän ydin. Automaatit on sijoitettu päämiesten tiloihin. Tällaisia tiloja ovat yleisesti kaupat, ostoskeskukset ja ravintolat. Automaattien käyttäjät ovat näiden toimipaikkojen asiakkaita. Käyttäjät ovat pelaajia, jotka sijoittavat tietyn panoksen automaattiin, pelaavat sen ja mahdollisesti saavat voiton tietyllä kertoimella. Rahapeliautomaatissa oleva peli voi olla joko mekaanisesti toimiva peli, videopeli tai näiden yhdistelmä. Kuvassa 2 on luonnos Barcrestin peliautomaatista.



Kuva 2. Rahapeliautomaatti

Kun automaatteihin laitetaan rahaa, ne menevät maksukassaan. Automaatti hyväksyy useanlaisia kolikoita sisään ja voitonmaksu tapahtuu myös useilla eriarvoisilla kolikoilla. Maksukassasta maksetaan mahdolliset voitot. Rahankerääjät ja huoltomiehet voivat myös täydentää maksukassaa, jos se on puutteellinen jonkin rahan osalta. Rahan sijasta tai yhdessä rahan kanssa automaatti voi ottaa myös vastaan pelimerkkejä.

Kun maksukassa täyttyy riittävälle tasolle, automaatti siirtää osan rahoista tyhjennyskassaan. Tyhjennyskassasta rahat eivät pääse takaisin pelikiertoon. Rahankerääjät tyhjentävät tyhjennyskassan riittävän väliajoin.

Rahapeliautomaatit sisältävät myös ohjelmamoduuleita, joilla voidaan säätää automaatin toiminnallisuutta. Eri moduulien toimintoja ovat mm. seuraavat asiat.

- Pelipanoksien arvot
- Voittoprosentti eli kuinka paljon automaatti jakaa voittoja suhteessa sisään laitettuun rahaan
- Testausohjelma, joka voidaan suorittaa huollon tai tarkastuksen jälkeen

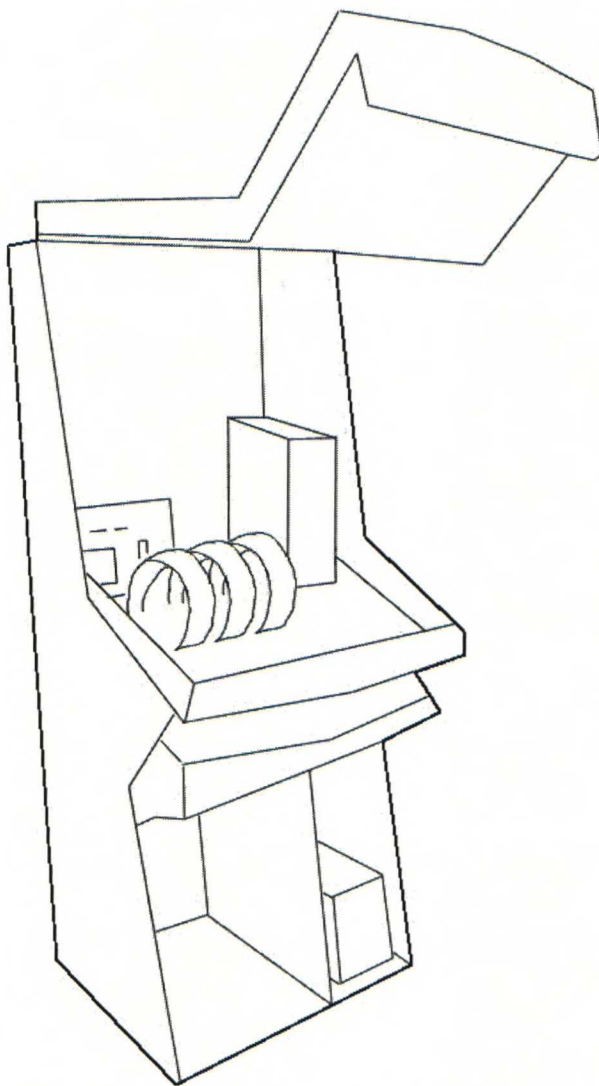
Rahapeliautomaatissa on laskureita, jotka pitävät tietoa erilaisten tapahtumien määrästä. Laskureita ei voi yleisesti ottaen nollata ja niiden arvoa voi vain kasvattaa. Käytännössä laskurit sisältävät arvoja rahojen liikkeistä. Alla olevassa listassa on kuvattu laskureiden mahdollisia tietoja.

- Rahaa sisään
- Pelimerkki sisään
- Rahaa maksettu voittona
- Pelimerkkejä maksettu voittona
- Rahan täydennys
- Pelimerkkien täydennys

Automaatin toimiessa Ison-Britannian alueella olisi käytössä seuraavat kolikot: 1p, 2p, 5p, 10p, 20p, 50p, £1, £2 sekä £5. Tällaisessa automaatissa on 54 rahalaskuria.

Laskureiden lisäksi automaattit keräävät tietoa kirjausketjuun. Merkintä kirjausketjussa sisältää aikaleiman sekä tapahtuman. Tapahtumat voivat olla tyypiltään hälytyksiä tai yleisiä tilannetietoja. Erilaisia tapahtumia on lueteltu seuraavassa listassa.

- Kolikoiden käsittelyjärjestelmässä häiriö
- Pelirullan vikaantuminen
- Valaistuksessa häiriö
- Painike jumiutunut
- Laskurin vikaantuminen
- EPROM-muistin virhe
- RAM-muistin tarkistuksessa virhe
- Muisti tyhjennetty
- Laitteistovika
- Jännitelähteen häiriö
- Staattinen sähkön purkaus havaittu
- Ohjelmamoduuli puuttuu
- Rahantäyttö tarpeellinen
- Tietyt kolikot loppu
- Huolto-oven avaus
- Huolto-oven sulkeminen
- Rahaoven avaus



Kuva 3. Rahapeliautomaatti avattuna

Kuvissa 2 ja 3 on hahmoteltuna Barcrestin rahapeliautomaatti. Barcrest on yritys, joka sijaitsee Yhdistyneissä kuningaskunnissa, mutta toimii Euroopan laajuisesti. Barcrest kuuluu Britannian peliautomaattiyrittäjien yhdistykseen BACTA:an (British Amusement Catering Trades Association). BACTA:an kuuluu tällä hetkellä yli 685 yritystä. /6, 7/

Kuvassa 3 on kuvattu rahapeliautomaatin tärkeimmät osat. Automaatissa on avattuna yläosassa huolto-ovi ja alaosassa rahaovi (ei näy kuvassa). Molemmat tilat ovat lukittu erillisillä avaimilla ja molempien avaus aiheuttaa merkinnän kirjausketjuun. Yläosassa on vasemmalla takana automaatin emolevy, jossa on keskusyksikkö, muistipiirit sekä ohjelmamoduulit. Koneen keskellä on mekaaniset pelikiekot. Maksukassa sijaitsee yläosan oikeassa laidassa. Rahakassa sijaitsee alaosan oikeassa laidassa, minkä kautta rahojen tyhjennys tapahtuu.

BACTA-protokolla määrää tapahtumien ja hälytyksien rakenteen, mitä voidaan seurata sarjaportista. Tietoliikenne jokaisessa BACTA-automaatissa on samanlainen. Sarjaporttiin voidaan liittää valvontayksikkö, joka toimii langattoman raportoinnin päätelaitteena. Valvontayksikön liittäminen automaattiin on määritelty esimerkiksi Markku Salinin diplomityössä ”Wireless remote monitoring system for a gaming machine”. /52/

Vaikka tässä kappaleessa on kuvattu Barcrestin BACTA-protokollaa käyttävän peliautomaatin toiminnallisuutta ja ominaisuuksia, ovat monet rahapeliautomaatit toiminnaltaan samankaltaisia. Maailmanlaajuisesti levinnyt espanjalaisen CIRSA on esimerkiksi tällainen. /10/

3 LIIKETOIMINNAN VAATIMUKSET

3.1 Yleiset vaatimukset

3.1.1 Yleistä

Liiketoiminnan puolelta tulee vaatimuksia langattomalle raportointijärjestelmälle. Järjestelmän avulla on kustannustehokkuutta sekä luotettavuutta parannettava. Uuden palvelun on myös mahdollistettava parempi tiedon saatavuus ja käsittely. Lisäksi järjestelmä ei saa olla estämässä tulevaisuuden liiketoimintasuunnitelmia.

Liiketoiminnan vaatimuksia on lueteltu seuraavaksi.

- Reaaliaikaiset hälytykset
- Säännöllinen tiedonkeruu
- Joustava ympäristö
- Kaksisuuntainen tietoliikenne
- Luotettavuus
- Luotettava raportointi
- Langattomat päätelaitteet

3.1.2 Reaaliaikaiset hälytykset

Täydellistä reaaliaikaisuutta järjestelmään ei tarvita. Riittää, että tietojenvälitykseen kuluva aika voidaan mitata minuuteissa, ei tunneissa. Raha-automaatin toiminnassa oloaika potentiaalisesti kasvaa, koska saadaan huoltopyynnot toimitettua nopeasti. Nopeaa tietoa luvattomasta rahakassaa suojaavasta rahaoven avaamisesta voidaan käyttää hyväksi varashälyttimenä.

3.1.3 Säännöllinen tiedonkeruu

Automaateista täytyy saada säännöllisin väliajoin tilastotietoa. Tiedot ovat käytännössä määrättyjen laskureiden lukemia. Kerran vuorokaudessa saatavat tiedot riittävät. Tiedonkeräyksen väli voi kuitenkin olla pienempi kuin vuorokausi.

3.1.4 Joustava ympäristö

Täytyy olla mahdollisuus siihen, että uusia automaatteja voidaan liittää järjestelmään jälkikäteen. Saatava tilastotieto voi olla laitteistokohtainen ja raportoitavat laskurit

voivat muuttua automaattikohtaisesti. On tarpeellista myös mahdollisuudelle laajentaa järjestelmää niin, että tietoa välitetään myös muihin järjestelmiin. Saman järjestelmän on kyettävä vastaanottamaan viestejä erilaisilta automaateilta, mutta ei kuitenkaan niin, että käytettäisiin eri protokollaa kommunikoimiseen.

3.1.5 Kaksisuuntainen tietoliikenne

Kaksisuuntaista tietoliikennettä tarvitaan, jotta voitaisiin lähettää komentoja myös automaattiin. Seuraavaksi on esitetty komentoja, joita voidaan lähettää automaattiin.

- Raportointiväli
- Raportoivat laskurit
- Tiedon pyyntö

Kaksisuuntaista liikennettä tarvitaan myös, jos halutaan tehdä seuraavia muutoksia.

- Pelikoneen voittomarginaalien asettaminen
- Ohjelmistopäivitys rahapeliautomaattiin
- Ohjelmistopäivitys tiedonkeruulaitteeseen

Päivitystoimenpiteet ovat raportointijärjestelmän ulkopuolella, mutta näiden toimintojen mahdollisuutta ei tule estää.

3.1.6 Luotettavuus

Raportoitava tieto täytyy saada riittävän luotettavasti. Raportointia voidaan käyttää hyväksi silloin, kun rahankerääjä on jokin kolmas osapuoli esimerkiksi automaatin päämies eli rahapeliautomaatin sijoituspaikan henkilökunta. Näin voidaan tarkistaa rahankerääjien ilmoittamien rahamäärien oikeellisuus. Jos jokin automaatti jättää raportoimatta tilastotietoja, siltä on voitava kysyä nämä tiedot uudelleen. Vaikka yksittäistä raportoimatta jäänyttä tietoa ei kysyttäisikään uudelleen, on tilastojen silti näytettävä oikein myöhemmin tulevien ilmoitusten jälkeen.

3.1.7 Langattomat päätelaitteet

Automaateilta kerättyä tietoa on kyettävä tarkastelemaan riittävällä tasolla myös mobiileilla päätelaitteilla. Näiltä päätelaitteilta on myös kyettävä lähettämään komentoja automaateille. Automaateilta tulevat hälytykset on voitava ohjata suoraan määrätyle huoltomiehelle. On myös voitava määrittää, mitkä hälytykset aiheuttavat

uudelleenohjauksen. Hälytystyypistä riippuen voi kynnys huoltomiehelle ilmoittamiseen olla määriteltävissä. Esimerkiksi vähintään viisi painikkeen jumiutumisia vuorokauden sisällä voi laukaista hälytyksen.

3.2 Henkilökunta ja käyttäjät

3.2.1 Yleistä

Henkilöresurssit aiheuttavat suuria kustannuksia, joten rahapeliautomaattien toimintaympäristön henkilökunnan työmäärää on kyettävä vähentämään.

3.2.2 Huoltomies

Huoltomiesten turhia käyntejä automaateilla on kyettävä vähentämään. Automaatin vikaantuessa on ilmoitus huollon tarpeesta tultava riittävän nopeasti huoltomiehille. On tarpeellista myös mahdollisuudelle ennakoida huoltokäyntejä.

3.2.3 Rahankerääjä

Mahdollisuus sille, että rahapeliautomaatin päämies voi toimia luotettavasti rahankerääjänä. Päämies on automaatin sijoituspaikassa oleva henkilökunta. Raportoinnin avulla voidaan tarkastaa rahankerääjien ilmoittamien rahamäärien oikeellisuus. Jos päämies toimii rahankerääjänä, poistuu rahankeräys kokonaan tai osittain operaattorin toimenkuvasta.

3.2.4 Käyttäjä

Raportointijärjestelmän ei pidä vaikuttaa suoraan rahapeliautomaatin käyttäjiin eli pelaajiin. Epäsuora vaikutus pelaajiin toimivalla raportointijärjestelmällä tulee olla lisääntyneenä automaatin toiminnassa oloaikana.

3.2.5 Päämies

Päämiehen roolia automaatin valvojana tulee voida vähentää, koska automaatti ilmoittaa automaattisesti vikaantumisista.

3.2.6 Operaattori

Rahapeliautomaattien operaattorin tulee saada liiketaloudellista hyötyä langattomasta raportointijärjestelmästä.

3.3 Automaattilaitteisto

Rahapeliautomaattiin tulee sarjaportin kautta liitettävä langaton raportointiyksikkö. Automaatti välittää tapahtumatiedot sekä laskureiden arvot raportointiyksikölle. Raportointiyksikkö välittää tiedot palvelimelle, jossa tehdään tiedon jatkokäsittely. Tiedonvälitys tapahtuu langatonta tietoliikenneverkkoa käyttäen automaatin ja palvelimen välillä. Tietoliikenne täytyy olla kaksisuuntainen, jotta automaatille voidaan lähettää komentoja.

3.4 Palvelinympäristö

Palvelimelta on yhdyskäytävä langattomaan tietoliikenneverkkoon, minkä kautta viestintä automaatin raportointiyksikön kanssa tapahtuu. Automaateilta saatu tieto täytyy tallentaa tietovarastoon. Tietovarastoon täytyy olla mahdollisuus myös tallentaa automaatin ja toimintaympäristön asetuksia. Palvelimen täytyy jatkokäsitellä saatua tietoa ja muodostaa siitä tilastoraportteja. Raportteja pitää päästä katsomaan myös mobiileista päätelaitteista. Palvelimen tehtäviin kuuluu myös välittää hälytyksiä huoltomiehille, jos automaateilta tulee määrättyjä tapahtumatietoja. Palvelinympäristöön täytyy olla mahdollista liittyä myös muista järjestelmistä.

4 TAPAHTUMAPOHJAISET JÄRJESTELMÄT

4.1 Yleinen toimintaperiaate

4.1.1 Yleistä

Tapahtumapohjainen kommunikointityyli on levinnyt laajalle. Tällä paradigmalla on suuri määrä sovellusalueita aina työpöytäsovelluksista laajalle levinneisiin aikakriittisiin järjestelmiin. Tällaisia ovat esimerkiksi liikenteen ohjausjärjestelmät, sähköinen kaupankäynti ja mobiili tietojenkäsittely. Vuonna 2004 jo 26:tta kertaa järjestettävän kansainvälinen ohjelmistoteknologian konferenssin yhteydessä järjestetään tapahtumapohjaisten järjestelmien työpaja eli workshop, jossa käsitellään näitä asioita. /26/

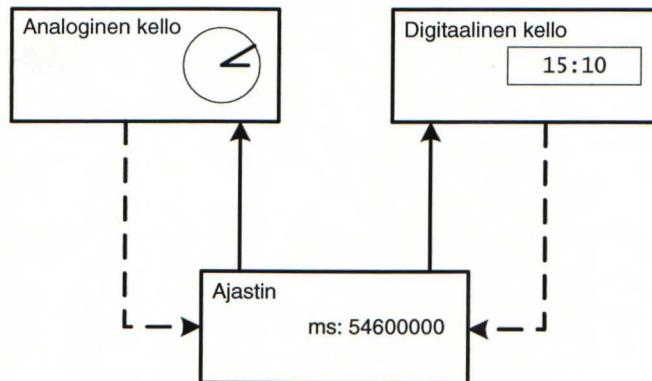
Tapahtumapohjaisissa järjestelmissä sovellukset lähettävät viestejä toisilleen. Yksittäisillä viesteillä voi olla yksi tai useampi vastaanottaja. Viestien vastaanottajat määräävät rajapinnan viesteille, jonka viestejä lähettävät sovellukset toteuttavat. Vastaanottava sovellus ei välttämättä tiedä lähettävän sovelluksen tyyppiä. Tälle tapahtumapohjaisten järjestelmien mallille on olemassa yleinen oliopohjainen suunnittelumalli, joka on nimeltään observer-malli.

4.1.2 Observer-malli

Observer-malli on oliopohjaisten hajautettujen tietojärjestelmien suunnittelumalli eli design pattern. Se määrittelee kaksi objektityyppiä, jotka ovat tarkkailija eli observer sekä kohde. Tarkkailija rekisteröi tietonsa kohdeobjektille, jolloin kohde tietää lähettää tilaansa koskevat muutostiedot tarkkailijalle. Kohteen ja tarkkailijan välinen suhde on yhdestä moneen, joten yhdellä kohteella voi olla useampi tarkkailija. Kohde ei tiedä tarkkailijansa tyyppiä, vaan näkee kaikki samaa tilanmuutosta tarkkailevat objektit samankaltaisina. /22/

Kuvassa 4 on esimerkki observer-mallista. Siinä on kohteena ajastin, jonka tila on aika millisekunteinä. Ajastimella on kaksi tarkkailijaa digitaalinen ja analoginen kello. Molemmat kellot ovat rekisteröityneet ajastimen tarkkailijaksi (katkoviiva). Ajan muuttuessa ajastin signaloi tiedon kelloille (kiinteä viiva). Ajastin näkee molemmat

kellot tarkkailijoina, jotka haluavat tiedon ajan muutoksista. Ajastin ei siis tiedä, että tarkkailijat ovat eri tyyppisiä.



Kuva 4. Esimerkki observer-mallista

Kohdeobjektin ja sen tarkkailijan välinen vuorovaikutus voi olla joko synkroninen tai asynkroninen. Synkronisessa toimintamuodossa lähettäjä jää odottamaan vastausta ennen toimintansa jatkumista. Asynkronisessa eli eriaikaisessa vuorovaikutuksessa lähettäjä jatkaa toimintaansa normaalisti saamatta suoraan vastausta kutsuun. /22/

4.2 Kommunikaatiojärjestelmät

4.2.1 Yleistä

Tapahtumapohjaisessa kommunikaatiojärjestelmässä asiakas- tai palvelinsovellus ilmoittaa palvelinsovellukselle, kun tietty tapahtuma tapahtuu. Tällaisia tapahtumia ovat pyytämättömät tapahtumatiedot odottamattomissa tilanteissa tai välitetyt tapahtumatiedot odottamattomana ajankohtana. Tapahtumapohjaisten järjestelmien paradigman laajamittainen hyväksyminen on johtanut de facto kommunikaatiojärjestelmiin ja standardeihin kuten CORBA ja JMS. JMS on kuvattu kappaleessa 7.4.2 J2EE. /26/

4.2.2 CORBA

CORBA eli Common Object Request Broker Architecture on Object Management Groupin avoin sovellusten keskinäisen verkon ylitse tapahtuvan kommunikoinnin arkkitehtuurin ja infrastruktuurin määrittely. CORBA:lla on mahdollista saavuttaa ohjelmien välille yhteensopivuus riippumatta sovelluksen toimittajasta,

käyttöjärjestelmästä tai ohjelmointikielestä. Tämä saavutetaan käyttäen hyväksi standardia IIOP-protokollaa (Internet Inter-ORB Protocol). /45/

4.2.3 Olemassa olevia järjestelmiä

Pankkien tilitapahtumat on esimerkki nykypäivän sovelluksesta, joka perustuu tapahtumapohjaiseen kommunikaatiojärjestelmään. Tilitapahtumien virtaviivainen käsittely, luotettavuus, tiedon eheys ja käyttöoikeus ovat keskeisiä ominaisuuksia järjestelmässä. Pankin päätelaite eli asiakassovellus suorittaa tilitapahtumat tekemällä transaktion pankkijärjestelmään eli palvelinsovellukselle.

4.3 Valvontajärjestelmät

4.3.1 Yleistä

Tärkeä sovellusala tapahtumapohjaisilla valvontajärjestelmillä on järjestelmien ja tietoverkkojen valvonta ja hallinta. Tässä mallissa verkon päätelaitteet tai ohjelmakomponentit signaloi määritellyistä, virheellisistä tai epänormaaleista tilanteista. Hälytykset välitetään hallinta-asemalle tai asemille, joissa hallinnoija muodostaa johtopäätökset sekä tekee oikeat ja tarvittavat toimenpiteet. Hallinnoija voi tässä tapauksessa olla hallintasovellus tai pääkäyttäjä. Automaattisesti käsiteltävät tapahtumien riippuvuudet ovat tärkeässä osassa tässä mallissa. Sarja hälytyksiä tai yksittäiset matalan tason hälytykset voivat johtaa korkean tason ongelmaan, jotka ratkaistaan virtana matalan tason tapahtumia. /26/

4.3.2 Olemassa olevia järjestelmiä

Pörssihälytykset ovat esimerkki tämän päivän tapahtumapohjaisesta valvontajärjestelmästä. Tiedon saannin nopeus pörssikurssin muutoksesta on keskeinen ominaisuus järjestelmässä. Asiakassovellus välittää parametrit eli osakkeen ja raja-arvon pörssijärjestelmään, joka toimii palvelinsovelluksena. Kun tietty raja-arvo ylitetään tai alitetaan, palvelinsovellus ilmoittaa tapahtumasta asiakassovellukselle.

5 TIETOLIIKENNE GSM-JÄRJESTELMÄSSÄ

5.1 Yleistä

GSM-matkapuhelinjärjestelmä on kansainvälinen standardi, jonka tekniikkaa on ollut kehittämässä johtavat kansainväliset yritykset. GSM-järjestelmä tarjoaa nopeat tiedonsiirtoyhteydet sekä rajoittamattoman liikkuvuuden.

5.2 Päätelaitte

Mobiiliasema eli GSM-verkon päätelaite koostuu fyysisestä laitteesta sekä ohjelmistosta, mitä käytetään kommunikointiin GSM-verkossa. /51/

Jokainen laite sisältää IMEI-koodin, joka on kansainvälinen mobiilitunniste. Tämän tunnisteen pohjalta puhelinoperaattorit voivat muodostaa valkoisia, harmaita ja mustia listoja. Valkoinen lista osoittaa, että päätelaiteen käyttö on sallittu. Musta lista sisältää sellaisten laitteiden identiteetin, joiden käyttö on kielletty. /17/

Laite sisältää lisäksi tilaajan tunnistusmoduulin eli SIM-kortin (Subscriber Identity Module). Tunnistusmoduuli sisältää tiedon liittymästä ja tämä esitetään IMSI-numerolla (International Mobile Subscriber Identity). Se koostuu kolme numeroa pitkästä maakoodista, kaksi numeroa pitkästä verkkotunnuksesta sekä kymmenen numeroa pitkästä tilaajanumerosta. MSISDN-numerolla (Mobile Station International ISDN), joka näkyy sekä A-tilaajan että B-tilaajan päässä, esitetään tämä sama numero ilman vakioipituutta. MSISDN-numero koostuu maatunnuksesta, verkkotunnuksesta ja liittymän tunnusnumerosta eli se on tilaajan puhelinnumero. /17, 51/

5.3 Lyhytsanomamat

Lyhytviestipalvelu (Short Message Service) mahdollistaa lyhytsanomien eli tekstiviestien lähettämisen GSM-verkossa. Tekstiviestit kulkevat GSM-verkossa tekstiviestikeskuksien (Short Message Service Center, SMSC) kautta. Lyhytviestipalvelu on määritelty ETSI:n standardeissa 03.38 ja 03.40. /16, 18/

Lyhytsanomaviestit käyttävät tiedonsiirtoon GSM-verkon signaalintikanavaa, joten se ei kuormita vakioviestikanavana käytettyä liikennöintikanavaa eli ns. puhekanavaa.

Lisäksi lyhytsanomat käyttävät vain signaalintikanavan käyttämätöntä kapasiteettia. Näin ollen tekstiviestien lähetyks ja vastaanotto voi tapahtua samaan aikaan, kun käynnissä on puheen tai datan lähetyks GSM-päätelaitteesta. /51/

Tekstiviestipalvelut koostuvat seuraavista viesteistä.

- SM MO on päätelaitteelta lähtevät tekstiviestit tekstiviestikeskuksen kautta
- SM MT on päätelaitteelle tulevat tekstiviestit tekstiviestikeskuksen kautta

Tekstiviestit voivat olla joko normaalisti tekstimuodossa tai PDU-formaatissa. PDU-formaatissa olevat viestit voivat käyttää mitä tahansa merkistön koodausmenetelmää.

Tekstiviestien sisältö on kooltaan 140 oktetia eli 1120 bittia. Käyttämällä koodaukseen 7 bittia merkkiä kohden saadaan viestiin mahtumaan 160 merkkiä. Unicode UCS2 –koodatut viestit sisältävät maksimissaan 70 merkkiä. /16, 18/

Tekstiviestejä on mahdollista myös ketjuttaa, joten voidaan muodostaa pitempiä viestejä kuin perusviesti. Ketjutetut viestit lähtevät ja liikkuvat GSM-verkossa kuitenkin erillään. Ketjuttaminen vähentää itse tiedon määrää yhdessä viestissä kuusi oktetia, koska ketjutusta koskevat tiedot ovat myös tietokentässä. Tekstiviestin tietokentässä (TP-User Data) pakkaamatonta 8-bittistä tietoa voi olla ketjutetuissa viesteissä 134 oktetia. 7-bittistä tietoa voi olla 153 merkkiä. Unicode UCS2-koodatuissa viesteissä on enimmillään 67 16-bittistä merkkiä. Viestejä voi olla ketjutettuna yhteensä 255 kappaletta. Kaavalla 1 voidaan laskea yksittäisen viestin pituus sekä kaavalla 2 ketjutetun viestin maksimi pituus. Taulukossa 2 on kuvattuna merkkien maksimimäärä yksittäisessä tai ketjutetussa viestissä koodaustavan mukaan. /16, 18/

Yksittäisen tekstiviestin pituus saadaan kaavalla 1.

$$n_1 = \frac{L \times o}{l} \quad (1)$$

jossa

L on yhden tekstiviestin koko oktetteina eli 140 oktetia,

o on oktetin koko bitteinä eli kahdeksan bittia oktetia kohden ja

l on yhden merkin koko bitteinä.

Ketjutetun tekstiviestin maksimi pituus saadaan kaavalla 2.

$$n_2 = N \times \left(n_1 - \left\lceil \frac{c \times o}{l} \right\rceil \right)$$

(2)

jossa
N on viestien maksimi määrä eli 255,
n₁ on yksittäisen viestin maksimi koko,
c on ketjutetun viestin määrittely oktetteina,
o on oktetin koko bitteinä eli kahdeksan bittiä oktettia kohden ja
l on yhden merkin koko bitteinä.

Merkistö	Yksittäisen viestin max. pituus	Ketjutetun viestin max. pituus
7-bittinen	160	39015
8-bittinen	140	34170
16-bittinen Unicode	70	17085

Taulukko 2. Merkkien määrä koodaustavan mukaan tekstiviestissä

5.4 Tiedonsiirto

Perus GSM käyttää tiedonsiirtoon piirikytkentäistä dataliikennettä (Circuit Switched Data). Vakio kaistanleveys tiedonsiirtoon on 9,6 kilobittiä sekunnissa. HSCSD-tekniikka (High Speed Circuit Switched Data) mahdollistaa kuitenkin nopeamman dataliikenteen. Tämä on toteutettu käyttäen useaa liikennöintikanavaa samaan aikaan. Tällä tiedonsiirtomenetelmällä voidaan saada enimmillään tiedonsiirtonopeudeksi 57,6 kilobittiä sekunnissa. Nopea HSCSD-tiedonsiirtotekniikka on ns. 2,5. sukupolven (2,5G) GSM-puhelimissa. /51/

Pakettikytkentäinen tiedonsiirtojärjestelmä (General Packet Radio Service, GPRS) on määritetty myös 2,5. sukupolven GSM-puhelimissa. Tämä menetelmä soveltuu hyvin pienten datamäärien säännölliseen siirtoon sekä keskisuurten datamäärien epäsäännölliseen siirtoon. Tällaisia ovat mm. HTTP-protokollalla tehdyt WWW-kyselyt ja -vastaukset. Tällä tiedonsiirtomenetelmällä voidaan saada enimmillään tiedonsiirtonopeudeksi 150 kilobittiä sekunnissa. GPRS-yhteyteen voidaan määritellä myös verkkoyhteyden ja palvelun laatu (Quality of Service, QoS). /51/

6 TIETOKANTAJÄRJESTELMÄ

6.1 Yleistä

Tässä luvussa tarkastellaan relaatiotietokantojen hallintajärjestelmää, tietokantojen käsittelykieltä sekä tietokantojen rajapintoja.

6.2 Relaatiotietokannat

Tri Edgard Frank Codd määritteli ensimmäisen kerran käsitteen relaatiotietokanta vuonna 1970. Relaatiotietokannassa tiedot esitetään kaksiulotteisina tauluina eli relaatioina. Yksi taulun horisontaalinen rivi muodostaa tietueen. Taulun rivit taas muodostuvat vertikaaleista sarakkeista eli kentistä. Taulukossa 3 on kuvattu relaatiomallin termistöä sekä niiden vastaavuutta tietokannan hallintajärjestelmässä sekä tiedostojärjestelmässä. /11/

Relaatiomalli	Tietokannan hallintajärjestelmä	Tiedostojärjestelmä
Relaatio (relation)	Taulu (table)	Tiedosto (file)
Monikko (tuple)	Rivi (row)	Tietue (record)
Attribuutti (attribute)	Sarake (column)	Kenttä (field)

Taulukko 3. *Relaatiotietokantamallin termistö /11/*

Relaatiotietokantaa mallinnetaan entity-relationship eli ER-mallilla. Tällä tietomallilla havainnollistetaan todellista maailmaa, joka sisältää perusobjekteja sekä näiden relaatioita eli suhteita toisiinsa. Perusobjekti eli entiteetti on sellainen olemassa oleva objekti, joka on erotettavissa muista objekteista. Perusobjekti rakentuu attribuuteista. Kuvassa 5 on esimerkki perusobjektista, joka kuvaa automaattia. Automaatti-objektilla on seuraavat attribuutit: valmistaja, nimi, tyyppi sekä sarjanumero.

<u>Objekti : Automaatti</u>
Valmistaja = Barcrest
Nimi = The Addams Family
Tyyppi = AWP
Sarjanumero = 684660001

Kuva 5. Esimerkki perusobjektista.

Relaatio kuvaa perusobjektien yhteenliittymistä ER-mallissa. Liittymiset voivat olla yhdestä yhteen, yhdestä moneen, monesta yhteen tai monesta moneen objektiin.

6.3 Tietokannan hallintajärjestelmät

6.3.1 Yleistä

Tässä kappaleessa tarkastellaan kolmen tietokannan hallintajärjestelmän (Database Management System, DBMS) keskeisiä ominaisuuksia. Oracle ja SQL Server ovat kaupallisia ohjelmia ja MySQL on GPL-lisenssillä oleva ilmainen tietokanta. Oracle-tietokannan ominaisuudet selvitetään ensin ja sen jälkeen sen oleellisimpia eroja SQL Serveriin ja MySQL:ään. Lopuksi tarkastellaan tietotyyppien eroja järjestelmien välillä.

6.3.2 Oracle Database

Oracle-tietokannasta on saatavilla eri versioita erilaisten tarpeiden mukaan. Enterprise Edition on tarkoitettu suurta suorituskykyä vaativiin toteutuksiin, jotka mahdollisesti vaativat klusterointia. Standard Edition on tästä hieman kevyempi versio, joka on tarkoitettu keskisuuriin toimintaympäristöihin. Tämä sisältää myös klusterointi mahdollisuuden, mikä puuttuu pieniin toimintaympäristöihin tarkoitettulta Standard Edition One -versiosta. Personal Edition on puolestaan tarkoitettu yksityiseen käyttöön niille, jotka tarvitsevat Oracle yhteensopivan tietokannan. Mobiileille laitteille on myös olemassa oma versio, joka on Lite Edition. Oracle-tietokanta on saatavilla mm. Sun Solaris, HP-UX, Linux ja Microsoft Windows käyttöjärjestelmiin.

On olemassa useita eri menetelmiä, joilla voidaan liittyä ohjelmallisesti Oracle-tietokantaan. Tietokannoille yleiset rajapinnat ODBC ja JDBC käsitellään omissa kappaleissaan.

OCI eli Oracle Call Interface on matalan tason ohjelmointirajapinta, joka suorittaa tietokantaoperaatioita kuten sisäänkirjautuminen, komentojen suoritus, jäsentäminen sekä tulosten hakeminen. OCCI eli Oracle C++ Call Interface on kehityskomponentti, joka perustuu OCI-ohjelmointirajapintaan sekä C++-kieleen. OCCI on tehokas ohjelmointirajapinta Oracle-tietokantaan. /46/

PL/SQL on Oraclen proseduraalinen kieli, joka on laajennus SQL standardiin. Kieli sisältää olioläheisen ohjelmoinnin tekniikoita. PL/SQL mahdollistaa useiden SQL-komentojen ryhmittämisen loogisesti. Näin saadaan parannettua suorituskkyä verrattuna siihen, että kutsuttaisiin jokaista SQL-komentoa erikseen ohjelmasta. /46/

Pro*C on Oracle-tietokantoja varten oleva sulautettu SQL C-kielelle. Sulautetussa SQL:ssä on mahdollista liittää SQL-komennot suoraan ohjelmointikieleen. Sulautettu SQL on laajennus itse ohjelmointikieleen ja sillä on oma esikäntäjä. /46/

Oraclen tärkeimpiä perustietotyypppejä langattoman raportointijärjestelmän kannalta ovat varchar2, char, number, date ja raw. Varchar2 on muuttuvan pituinen merkkijono, jolle määritellään maksimi pituus. Oraclessa tyhjä merkkijono on sama asia kuin määrittelemätön merkkijono eli NULL-arvo. Char on kiinteäpituinen merkkijono. Number on desimaaliluku, jolle määritellään merkitsevän numeron pituus sekä mittakaava. Tietotyyppi date sisältää päivämäärän sekä kellonajan. Tietotyyppiin raw voidaan tallentaa binääritietoa kuten salattua tietoa tai kuvia. /46/

6.3.3 SQL Server

SQL Server on Microsoftin tietokannan hallintajärjestelmä. Se on saatavissa eri Windows-käyttöjärjestelmille. /35/

Transact SQL eli T-SQL on laajennus ANSI SQL-kieleen. Se on dynaaminen tietokannan ohjelmointikieli. Oraclen PL/SQL on samankaltainen laajennus kuin T-

SQL, mutta ne eroavat kuitenkin toisistaan sen verran ettei suora konversio näiden välillä ole mahdollista. /35/

SQL Serverin tärkeimpiä perustietotyyppejä langattoman raportointijärjestelmän kannalta ovat varchar, char, number, decimal, datetime ja varbinary. Varchar on muuttuvan pituinen merkkijono, jossa tyhjä merkkijono on eri asia kuin NULL-arvo. Number on kokonaisluku ja decimal on desimaaliluku. Datetime sisältää päivämäärän sekä kellonajan. Binääritietoa voidaan tallentaa kenttään, jonka tietotyyppi on varbinary. /35/

6.3.4 MySQL

MySQL-relaatiotietokanta pohjautuu avoimeen lähdekoodiin ja on saatavissa useisiin eri käyttöjärjestelmiin kuten yleisimmät Linux-jakelut, Mac OS, UNIX sekä Microsoft Windows. MySQL on saatavilla GPL-lisenssiehdoilla (GNU General Public License) sekä kaupallisella lisenssillä. Tietokanta tukee kuitenkin transaktiota vasta versiosta 4.0 lähtien. /37/

Avoimen lähdekoodin GPL-lisenssi määrittelee mm. seuraavia asioita: ohjelmasta saa olla asennuksia ja käyttäjiä rajaton määrä sekä ohjelmiston muokkausta ja jakelua ei ole rajoitettu millään tavalla. /23/

Kaupallinen lisenssi tarvitaan MySQL-ohjelmistoon, jos sitä jaetaan ohjelman mukana, mikä ei perustu avoimeen lähdekoodiin. Tämä tarvitaan, jos halutaan valtuutus MySQL Ab -yhtiöltä ohjelmistoon tai halutaan tukipalveluita kehitykseen. /37/

Tärkeimmät ajurit, joilla voidaan liittyä MySQL-tietokantaan ovat MySQL Connector/J, MySQL Connector/ODBC sekä MySQL Connector/Net. Nämä toteuttavat JDBC-, ODBC- sekä ADO.NET-ohjelmajapinnat. /38/

MySQL:n tärkeimpiä perustietotyyppejä langattoman raportointijärjestelmän kannalta ovat varchar, char, int, decimal, datetime, tinyblob. Varchar on muuttuvan pituinen merkkijono, missä tyhjä merkkijono on eri asia kuin NULL-arvo. Int on kokonaisluku

ja decimal on desimaaliluku. Datetime sisältää päivämäärän sekä kellonajan. Tinyblob ja blob ovat binääritietotyyppejä. /37/

6.3.5 Tietotyypit

Taulukossa 4 on kuvattu raportointijärjestelmän kannalta oleellisia tietotyyppejä. Tietotyypit on esitetty JDBC:n, Java-kielen ja eri tietokantojen välillä. Tietokantatyyppien maksimikoot vaihtelevat eri tietokantojen välillä, vaikka muuten ovat samankaltaisia.

JDBC	Java	Oracle	SQL Server	MySQL
VARBINARY	byte[]	RAW	VARBINARY	BLOB
INTEGER	int	NUMBER	NUMBER	INT
NUMERIC	java.math.BigDecimal	NUMBER	DECIMAL	DECIMAL
VARCHAR	String	VARCHAR2	VARCHAR	VARCHAR
TIMESTAMP	java.sql.Timestamp	DATE	DATETIME	DATETIME
CHAR	String	CHAR	CHAR	CHAR

Taulukko 4. Tietotyypit eri järjestelmissä

6.4 SQL-kieli

SQL-kieli eli Structured Query Language on yleisesti käytössä oleva relaatiotietokantojen käsittelykieli. Alunperin SQL-kieli oli nimellä SEQUEL (Structured English QUery Language). SQL-kielestä on monta versiota, joista ensimmäisen standardin määritteli ANSI vuonna 1986, mitä yleisesti kutsutaan SQL1:ksi tai SQL-86:ksi /1/. Seuraavana vuonna ISO määritteli myös saman standardin. Vuonna 1989 uusi standardi SQL-89 korvasi SQL-86 –standardin /2, 29/. Vuonna 1992 SQL-kielestä tuli versio, joka määrittelee itse kielen lisäksi myös sen miten se sulautuu eri ohjelmointikieliin. Tätä versiota usein kutsutaan SQL2:ksi tai SQL-92:ksi. Standardin on määritellyt kaksi tahoa, kuten sen edellisetkin versiot /3, 30/. Vuonna 1999 julkaistu SQL3 tai SQL-99 –standardi määrittelee oliolähtöisen relaatiotietokantamallin /4, 31/.

Vaikka SQL-kieli on standardi, useimmissa tietokannan hallintajärjestelmissä on sovelluskohtaisia laajennuksia ja osalle toiminnoista on eriävä syntaksi. Tällaisia laajennuksia ja eroavaisuuksia ovat mm. funktiot ja liitokset kyselyissä. Kaikki

järjestelmät eivät myöskään toteuta kaikkea sitä, mikä on määritelty standardissa. Tietty SQL-koodi ei välttämättä toimi toisessa tietokannan hallintajärjestelmässä kuin siinä mihin se on kirjoitettu.

SQL-kielellä on mahdollista tehdä mm. seuraavia asioita tietokannan hallintajärjestelmiin:

- tietokantakäyttäjien valtuuksien asettaminen,
- tietokannan rakenteen määrittäminen,
- rivien lisäys,
- rivien päivittäminen,
- rivien poistaminen ja
- kyselyjen tekeminen.

6.5 Rajapinnat

6.5.1 Järjestelmäkohtaiset rajapinnat

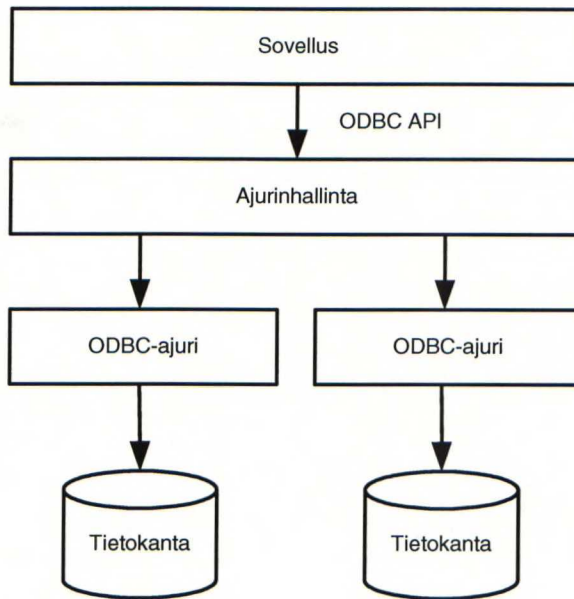
Järjestelmäkohtaiset eli natiivit rajapinnat ovat tietokantajärjestelmästä sekä ohjelmointikielestä riippuvia. Järjestelmäkohtaisilla rajapinnoilla muodostetaan suora yhteys ohjelmakomponentista tietokantaan.

6.5.2 ODBC

ODBC eli Open Database Connectivity on tietokannan ohjelmointirajapinta (API). ODBC on Microsoftin määrittely ja se rakentuu ISO:n Call-Level Interface /28/ ja X/Open:in SQL Call Level Interface /76/ standardien päälle. ODBC:n kolmas versio toteuttaa täysin molemmat standardit. Aikaisemmat versiot ohjelmointirajapinnasta eivät toteuttaneet täysin molempia standardeja. /34/

Rajapinta on riippumaton tietokannan hallintajärjestelmästä tai käyttöjärjestelmästä. ODBC-ajurien ei tarvitse toteuttaa täyttä SQL-92 tasoa, mutta niiden täytyy toteuttaa minimitaso, joka on osajoukko SQL-92:sta. /34/

ODBC-arkkitehtuuri määrittelee 4 tasoa: sovellustaso, ajurinhallinta, ajuri sekä tietolähde. Nämä tasot on esitetty kuvassa 6.



Kuva 6. ODBC-arkkitehtuurin tasot /34/

Sovellustaso suorittaa SQL-lauseiden käsittelyn ODBC:n funktioilla sekä vastaanottaa tulokset. Sovellukset jakautuvat kolmeen kategoriaan: yleiset sovellukset, vertikaalisovellukset sekä asiakaskohtaiset sovellukset. Yleisiä sovelluksia ovat esimerkiksi sovelluskehitysympäristöt. Vertikaalisovellukset suorittavat yksinkertaisia tehtäviä kuten tilausten syöttö tai tuotannon seurannan tiedonkeruu. /34/

Ajurinhallinta lataa ja poistaa ajurit sovelluksen puolesta. Tämä suorittaa ODBC-funktiokutsut ja välittää ne ajurille. /34/

Ajuri suorittaa ODBC-funktiokutsut, lähettää SQL-kyselyt tiettyyn tietolähteeseen sekä palauttaa tulokset sovellukselle. Ajuri tekee myös tarvittaessa sovelluksen kutsuille syntaksimuutokset tiettyä tietokannan hallintajärjestelmää varten. /34/

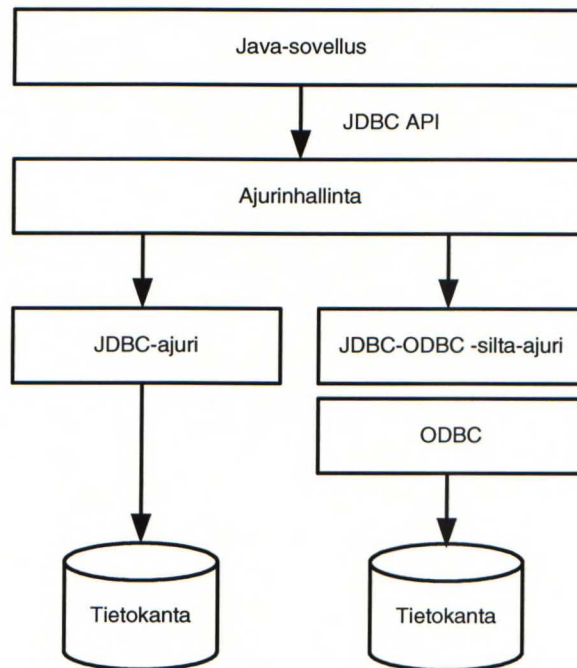
Tietokerros sisältää tiedon siitä mihin halutaan päästä käsiksi ja siihen kuuluvan käyttöjärjestelmän, tietokannan hallintajärjestelmän ja mahdollisen käytettävän verkkoalustan. /34/

6.5.3 JDBC

JDBC eli Java Database Connectivity on ohjelmointirajapinta tietokannan ja Java-ohjelman välillä. Se mahdollistaa liitettävyyden myös muihin tietolähteisiin kuin

relaatiotietokantoihin. Tällaisia tietolähteitä ovat taulukkopohjaiset tietolähteet kuten taulukkolaskentaohjelmat ja rakenteettomat tiedostot. /15/

On myös olemassa JDBC-ODBC –siltoja, jonka avulla voidaan liittyä tietokantoihin JDBC:llä, mitkä tukevat vain ODBC-rajapintaa. JDBC-arkkitehtuuri on esitetty kuvassa 7. /15/

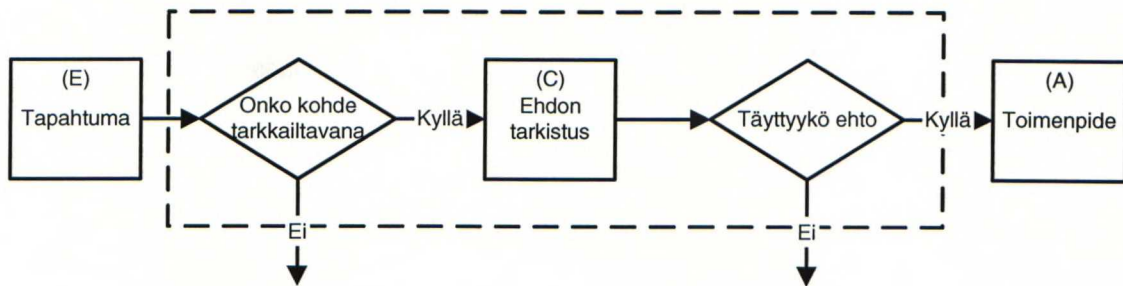


Kuva 7. JDBC-arkkitehtuuri /15/

6.6 Tietokannan laukaisimet

Tietokannan laukaisimilla (trigger) voidaan automaattisesti reagoida muutoksiin tietokannassa. Tietokannan hallintajärjestelmiä, jotka voivat reagoida niissä tapahtuviin muutoksiin, kutsutaan aktiivisiksi tietokannoiksi. Passiivisissa järjestelmissä tämä logiikka täytyy toteuttaa erillisillä sovelluksilla. /70/

Yleinen ECA eli event-condition-action sääntö, jota käytetään tietokannan laukaisimissa, on esitetty kuvassa 8. Tapahtuma (event) on jokin tietokannan toimenpide kuten päivitys (update) tai lisäys (insert). Ehdon (condition) tarkistus perustuu boolean logiikkaan. Toimenpide (action) voi olla esimerkiksi proseduurikutsu tai jokin toimenpide. /70/



Kuva 8. ECA-sääntö tietokannan laukaisimessa /70/

7 PALVELIMET

7.1 Yleistä

Tässä luvussa esitetään langattomassa raportointijärjestelmässä käytettävät palvelimet lukuunottamatta tietokantapalvelinta, jolle on oma lukunsa. Lisäksi selvitetään palvelinten välisiä rajapintoja sekä ohjelmointi-, komento- sekä julkaisukieliä.

7.2 SMS-yhdyskäytävä

SMS-yhdyskäytävä (SMS Gateway) toimii yhdyskäytävänä SMSC:n eli tekstiviestikeskuksen ja ohjelmistojärjestelmän välillä. SMS-yhdyskäytävän avulla voidaan lähettää ja vastaanottaa SMS-viestejä.

Tulevat viestit menevät jonoon, josta niitä voidaan sovellusohjelman avulla lukea. Erilaisia ohjelmarajapintoja SMS-yhdyskäytävän ja sovellusohjelmiston välissä ovat mm. HTTP, XML sekä RCP (Remote Call Procedure).

SMS-yhdyskäytäväratkaisuja löytyy useilta eri toimittajilta. Exomi Messaging Gateway, johon on liitetty SMS Gateway -moduuli, on eräs ratkaisu. /19/

7.3 WWW-palvelin

7.3.1 Yleistä

WWW-palvelimet vastaanottavat HTTP-yhteyksiä. Yhteydenotto tapahtuu selaimista, joihin palvelin toimittaa WWW-sivut. Tunnettuja WWW-palvelimia ovat mm. Microsoft Internet Information Services /33/, Sun Java System Web Server /65/ ja Apache /5/. Apache HTTP Server Project perustuu avoimeen lähdekoodiin.

7.3.2 CGI

CGI eli Common Gateway Interface on WWW-palvelinvalmistajien sopima rajapinta, jonka avulla saadaan palvelimeen integroitua skriptejä tai ohjelmia. Tämä mahdollistaa dynaamisen informaation luomisen. /32/

CGI-rajapinta on riippumaton ohjelmointikielestä. CGI-ohjelmat voidaan olla kirjoitettu kaikilla kielillä mitä voidaan ajaa järjestelmässä kuten C/C++, Perl tai

Fortran. Rajapinta määrittää miten WWW-palvelin välittää tiedon jollekin ohjelmalle ja mitä tietoa välitetään. /32/

7.3.3 PHP

PHP on Apache Software Foundationin projekti, joka pohjautuu vapaaseen lähdekoodiin. PHP mahdollistaa palvelinpuolen WWW-ohjelmoinnin (server-side scripting). Tämän avulla voidaan tehdä dynaamisia HTML-sivuja sulauttamalla PHP-tulkki WWW-palvelimeen. /48/

PHP on saatavilla moniin eri käyttöjärjestelmiin kuten Solaris, HP-UX, Windows 98/ME ja Windows NT/2000/XP. PHP tukee yleisiä WWW-palvelimia, joita ovat Apache, Microsoft Internet Information Services (IIS) sekä Sun Java System Web Server. PHP on mahdollista asentaa myös monille muille alustoille, koska sen mukana tulee lähdekoodi, jonka voi kääntää haluamaansa ympäristöön. /48/

PHP tukee monia yleisesti käytettäviä tietokantoja ja tietokantarajapintoja kuten ODBC, Oracle (OCI7 ja OCI8), IBM DB2 ja MySQL. PHP:ta on mahdollista käyttää myös komentoriviskriptien (commandline scripting) ajamiseen. /48/

7.3.4 HTML

HTML (HyperText Markup Language) on lingua franca eli yhteinen kieli hypertekstien julkaisuun. HTML pohjautuu SGML-standardiin ja sillä kuvataan tekstin rakennetta. Rakenteen merkitseminen tapahtuu HTML-elementeillä. Rakenteen lisäksi HTML:llä voidaan myös kuvata ulkoasua, mutta sitä varten on myös olemassa tyylimäärittelyt eli CSS, joka voidaan linkittää suoraan HTML-dokumenttiin. /27, 73/

XHTML (Extensible Hypertext Markup Language) on dokumenttien merkkauškieli, joka on sekä laajennus että alijoukko HTML:stä. XHTML on myös XML-dokumentti. XHTML Basic on suppeampi versio merkkauskielestä ja se on tarkoitettu lähinnä mobiileille päätelaitteille. /74, 75/

7.3.5 WML

WML (Wireless Markup Language) on mobiileja laitteita varten suunniteltu merkkaukieli. Tällaisia laitteita ovat erityisesti WAP-päätelaitteet. WML pohjautuu XML:stä ja on erittäin suppea, jos sitä vertaa HTML:ään. WML ei käytännössä sisällä ulkoasun määrittäksiä. /69/

Yksi WML-sivu on kortti (card) ja sivusto on pakka (deck). Pakka ladataan aina kerralla päätelaitteeseen. Päätelaite voi kuitenkin rajoittaa pakan suurinta kokoa. /69/

7.3.6 CSS

CSS (Cascadin Style Sheets) on erityisesti WWW-dokumenttien ulkoasun tyylien määrittämiseen tarkoitettu ohje. Kun HTML-merkkaukieli kuvaa dokumentin rakennetta ja sisältöä, CSS kuvaa miten dokumentti tulisi esittää.

CSS:stä on tehty kaksi määrittystä. Ensimmäinen taso sisältää määrittelyt mm. fonttien, marginaalien ja värien ominaisuuksille. Toinen taso sisältää kokonaan ensimmäisen tason ominaisuudet sekä määrittelee lisäksi mm. elementtien absoluuttinen paikanmäärittäminen, sivukatkokset, automaattisen numeroinnin ja oikealta vasemmalle olevan tekstin. /71, 72/

CSS:n taso kolme on tällä hetkellä kehitteillä. On myös tehty toisen tason ja kolmannen tason osajoukkoja kuten Mobile Profile, Print Profile ja TV Profile.

7.4 Sovelluspalvelimet

7.4.1 Yleistä

Sovelluspalvelimet koostuvat ohjelmakomponenteista, joissa suoritetaan sovelluksen looginen toiminnallisuus eli business-logiikka. Sovelluslogiikka sovelluspalvelimissa on yleensä toteutettu Java-kielellä. Sovelluspalvelimet voivat sisältää tai sulautua tietokantapalvelimeen ja WWW-palvelimeen.

7.4.2 J2EE

Java 2 Platform, Enterprise Edition eli J2EE määrittelee käytännöt monisäikeisille komponenttipohjaisille Java-sovelluksille. Nämä käytännöt sisältävät mm. EJB-

komponenttien, Java Message Service –ohjelmistorajapinnan sekä Java Naming and Directory Interface –nimipalvelun määritelmät.

EJB eli Enterprise JavaBean –teknologia on J2EE-ympäristön komponenttiarkkitehtuuri. Se mahdollistaa hajautetun, transaktionaalisen, turvallisen sekä siirrettävän ohjelmistokehityksen. /13/

JMS eli Java Message Service on viestintärajapinta, joka mahdollistaa viestien luonnin, lähettämisen, vastaanottamisen ja lukemisen Java-komponenteissa. Tämä viestintärajapinta mahdollistaa hajautetun viestinvälittämisen, joka on asynkroninen, luotettava ja löyhästi kytketty. /24/

JNDI eli Java Naming and Directory Interface tarjoaa yhdistetyn rajapinnan useisiin nimeämis- ja hakemistopalveluihin. Tämä mahdollistaa tiedon välityksen eri käyttäjien, tietokoneiden, tietoverkkojen, palveluiden sekä sovellusten välillä. /62, 63/

J2EE-sovelluspalvelimia ovat mm. Sun Java System Application Server /64/, BEA WebLogic Server /8/, IBM WebSphere Application Server /25/ sekä Oracle Application Server /47/.

7.5 Rajapinnat

7.5.1 Yleistä

Tässä kappaleessa tarkastellaan rajapintoja, joita käytetään tiedon siirtämiseen raportointiyksiköltä sovelluspalvelimelle. Näitä ovat OTA, HTTP ja RPC. OTA-rajapintaa käytetään tiedon siirtämiseen raportointiyksikön ja sovelluspalvelimen välillä. HTTP- tai RPC-rajapintoja käytetään OTA-viestin siirtämiseen SMS-yhdyskäytävän sekä sovelluspalvelimen välillä. SMS-yhdyskäytävästä riippuen myös muut vastaavanlaiset rajapinnat ovat mahdollisia.

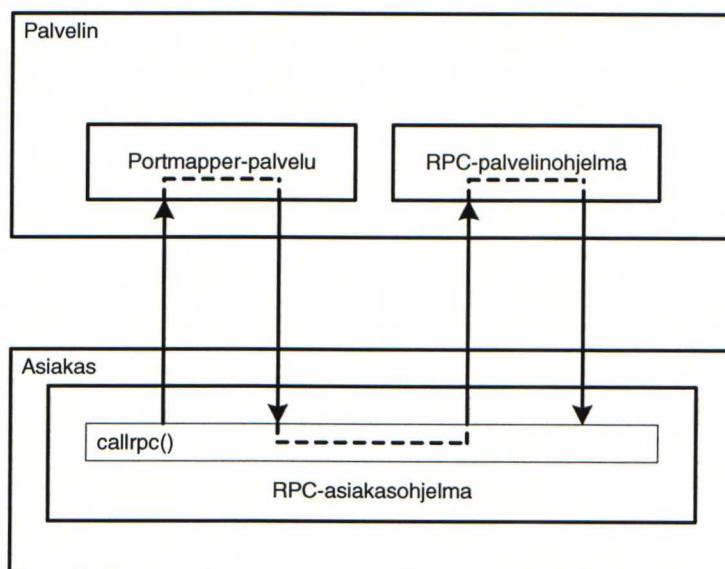
7.5.2 RPC-etäproseduurikutsu

ONC RPC –etäproseduurikutsu eli Open Network Computing Remote Procedure Call on menetelmä, jolla asiakas-palvelinmallissa palvelun pyytäjä pystyy kutsumaan palvelinta. Palvelin rekisteröi RPC-sovelluksen ohjelma-, proseduri- sekä

ohjelmanumeron. Asiakasohjelma kutsuu tietyn palvelimen RPC-palvelua näitä numeroita käyttäen. Tietyllä ohjelmanumerolla ei voi olla kuin yksi RPC-palvelu palvelimessa ja se on määriteltävissä heksadesimaalilukujen 20000000 ja 3fffffff välille. /61/

Tietty palvelu voidaan määritellä RPC-protokollan kuvauskielellä (RPC Language). RPCGEN on puolestaan protokollakääntäjä, joka tulkitsee kuvauskielen ja muodostaa palvelimen (server stub/skeleton) sekä asiakasohjelman ohjelmarunгон (client stub) sovelluksille. Kuvauskielellä määritellään proseduurin vastaanottamat argumentit sekä palauttavat parametrit. Lisäksi kuvauskielellä määritellään ohjelma-, proseduri- sekä ohjelmanumerot. /9/

RPC-protokolla ei määrittele tiedonsiirtoprotokollaa, mutta usein RPC-palvelinohjelma rekisteröi itsensä palvelimen portmapper-palvelulle, joka kuuntelee TCP-porttia 111. Asiakasohjelma kutsuu ensin portmapper-palvelua paikallistaakseen palvelinohjelman ja tekee sen jälkeen etäproseduurikutsun itse palveluun. Etäproseduurikutsun muodostuminen on kuvattu kuvassa 9. /9, 61/



Kuva 9. *RPC-kutsun muodostuminen /9/*

Jos langattomassa raportointijärjestelmässä SMS-yhdyskäytävä tukee RPC-rajapintaa, SMS-yhdyskäytävä sekä M2M-palvelinohjelmisto toimivat molemmat RPC-asiakasohjelmana sekä RPC-palvelimena. Vastaanotettaessa viestejä

raportointiyksiköltä SMS-yhdyskäytävä toimii asiakasohjelmalla. Kun taas lähetetään viestejä raportointiyksikölle, M2M-palvelin toimii asiakasohjelmalla ja SMS-yhdyskäytävä palvelinohjelmalla.

7.5.3 HTTP

HTTP-protokolla on yleinen tiedonsiirtoprotokolla TCP/IP yhteyksissä. Protokollan oletus TCP-portti on 80. HTTP-viestit käsittävät kyselyt asiakassovellukselta sekä vastaukset palvelimelta asiakkaalle. Kyselyt ja vastaukset voivat sisältää otsikon ja viestin rungon. HTTP-kysely muodostuu palvelupyynnöstä, joita ovat OPTIONS, GET, HEAD, POST, PUT, DELETE, TRACE ja CONNECT. Otsikkotiedoilla voidaan parametrisoida kyselyä tai kuvata vastauksen sisältöä. /20/

Vastausviesti sisältää kolminumeroisen tilatiedon, joka kertoo palvelimen kyvyn ymmärtää kysely sekä kyvyn toteuttaa palvelupyyntö. Alla on esitettyinä muutamia tilatietoja.

- 200 OK – Kysely on onnistunut
- 400 Bad Request – Palvelin ei kyennyt tulkitsemaan kyselyä
- 403 Forbidden – Kysely ymmärretty, mutta hylätty
- 404 Not Found – Palvelin ei löytänyt kyselyä vastaavaa dokumenttia
- 500 Internal Server Error – Palvelimessa tapahtui odottamaton tila

/20/

Jos langattomassa raportointijärjestelmässä SMS-yhdyskäytävä tukee HTTP-rajapintaa, SMS-yhdyskäytävä ja M2M-palvelinohjelmisto voivat välittää SMS-viestejä toisilleen GET- ja POST-kyselyiden avulla. Kuvassa 10 on esimerkki POST-viestistä, jossa kutsutaan SMSMessage-sivua. Siinä määritellään HTTP-protokollan otsikkotiedot Host, Content-Type ja Content-Length. Lisäksi määritellään viestin runko, joka sisältää parametrit MSISDN-numerolle ja SMS-viestille.

```

POST /SMSMessage HTTP/1.1
Host: M2Mserver
Content-Type: application/x-www-form-urlencoded
Content-Length: 44

MSISDN=1234567890&Message=Testi+viesti

```

Kuva 10. Esimerkki POST-viestistä

7.5.4 OTA

OTA eli Over The Air on yleisnimitys mobiilien päätelaitteiden käyttämille rajapinnoille. Se voi käsittää protokollan tai tietosisällön kuvauksen. Tässä kappaleessa keskitytään tietosisältöön, koska järjestelmän on tarkoitus käyttää GSM-verkon tiedonsiirtopalveluita.

Markku Salinin Over The Air Interface –dokumentissa on määritelty OTA-rajapinta tapahtuma- ja tilastoviestien rakenteelle rahapeliautomaateista. Molemmat viestityypit sisältävät otsikko- ja sisältöosan. Otsikkotiedoissa on tietoa lähettäjästä ja itse viestistä. /53/

Tapahtumaviestin sisältöosa käsittää tapahtuman tunnisteiden ja mahdollisen arvon. Tapahtuman tunniste on kolmimerkkinen koodi, joka kuvaa tapahtumaa. Arvo on raportoitavan tapahtuman mahdollisen laskurin arvo. /53/

Tilastoviesti sisältää raportoitavien ajankohtien määrän viestissä, raportointi ajankohdat, laskureiden tunnisteiden sekä niiden arvot. Yksiviesti voi siis sisältää useamman laskurin arvon eri ajankohtina. Raportoinnin ajankohdat on merkitty viikonpäivä ja kellonaika pareina. /53/

7.6 Salausalgoritmit

7.6.1 HTTPS

HTTPS on HTTP-protokollan versio, jolla voidaan muodostaa salattu tietoliikenneyhteys. HTTPS käyttää salaukseen SSL- tai TLS-protokollaa. TLS /14/ (Transport Layer Security) on SSL:n /49/ (Secure Sockets Layer) kehittyneempi versio. Näitä protokollia käytetään myös muiden kuin HTTP-tietoliikenneyhteysien

salaamiseen. Tällaisia ovat sähköpostiprotokollat IMAP /12/ (Internet Message Access Protocol) ja POP /36/ (Post Office Protocol) sekä hakemistoprotokolla LDAP /66, 67/ (Lightweight Directory Access Protocol). HTTPS-protokollan käyttämä oletus TCP-portti on 443.

7.6.2 MD5

MD5 on professori Ronald L. Rivestin kuvaama tiivistealgoritmi (message digest), mikä tuottaa 128-bittisen tiivisteen. MD5-algoritmia käytetään tiedon tarkistesummana sekä salasanojen tallentamiseen. Tiivistealgoritmien turvallisuus perustuu siihen, ettei tiivisteestä pystytä muodostamaan lähdettä. Tiivistealgoritmit ovat siis yksisuuntaisia funktiota. /50/

8 M2M-JÄRJESTELMIÄ JA -LAITTEITA

8.1 Yleistä

Tässä luvussa käydään kahden laitevalmistajan, Nokian ja Siemensin, M2M-liiketoimintaan suunniteltuja laitteita ja ratkaisuja.

8.2 Nokia M2M Platform

8.2.1 Yleistä

Nokia M2M –alusta sisältää päätelaitteita, moduuleita, sovelluskehitysalustan sekä yhdyskäytäväpalvelimen. Alusta mahdollistaa eri GSM-verkkopalveluiden käyttämisen parametrisoimalla päätelaite sekä yhdyskäytäväpalvelin. Tällaisia verkkopalveluita ovat SMS, GSM data, USSD sekä GPRS.

Nokia M2M –alusta perustuu CORBA-teknologiaan. Tämä tarkoittaa sitä, että etäsovellukset sekä palvelinsovellukset tekevät funktiokutsuja toisilleen. Funktiokutsut määritellään CORBA IDL:llä eli rajapintojen määrittelykielellä.

Koko alustan perustana on yhdyskäytäväpalvelimena toimiva Nokia M2M Gateway, joka toimii tiedonvälittäjänä GSM-verkon ja Internetin välillä. Yhdyskäytäväpalvelin välittää esimerkiksi sovelluspalvelimelta saadun funktiokutsun SMSC-tekstiviestikeskuksen kautta Connectivity Terminal –pätelaitteelle, josta se ohjautuu etäsovellukselle.

8.2.2 Sovelluskehitysalusta

Nokia M2M Application Development Kit (ADK) on kehitysympäristö, joka tarjoaa ohjelmistokehitystyökaluja sovelluksien rakentamiseen ja testaamiseen. ADK sisältää mm. Connectivity Terminal –pätelaitteet, analysointiyksikön (evaluation module), IDL-kääntäjän sekä yhdyskäytäväohjelmiston. /40/

Analysointiyksikkö tukee CORBA-viestejä ja se sisältää kahdeksan digitaalista input- ja output-porttia. Yksikössä on myös RS-232 –sarjaliikenneportti sekä digitaalinen audioliitäntä (digital audio interface, DAI). /40/

8.2.3 Päätelaitteet

Nokia tarjoaa päätelaitteita sekä moduuleita langattomiin M2M-ratkaisuihin. Näitä ovat Nokia 12 ja 12i –moduulit /41, 42/ sekä Nokia 30 ja 31 –päätelaitteet /43, 44/. Moduulit ja päätelaitteet tukevat GSM-verkon verkkopalveluita kuten SMS teksti- ja dataviestejä, GSM-dataa, USSD:tä sekä GPRS:ää.

8.3 Siemens M2M

Siemens tarjoaa useita GSM/GPRS-moduuleita ja –päätelaitteita langattomaan viestinvälitykseen useille liiketoimialoille. Siemens Wireless Community –ohjelman jäsenyritykset tarjoavat sovelluksia mm. mittaukseen, etävalvontaan, asuinturvaan, logistiikkaan, autoihin sekä myyntiautomaatteihin. Alla on esitetty muutamia Siemensin langattomaan liiketoimintaan tarkoitettuja moduuleita ja päätelaitteita sekä niiden erityispiirteitä ja sovelluksia.

- Wireless Module XT55 –moduulissa on sisäänrakennettu GPS-vastaanotin satelliittinavigointia varten. Turvallisuus, kuljetus ja logistiikka ovat soveltuvia liiketoimialoja. /60/
- Wireless Module TC45 –moduulissa on J2ME Information Module Profile tuki sekä yhdeksän ohjelmoitavaa GPIO-porttia (General Purpose Input/Output). Moduuliin on mahdollista tehdä ohjelmapäivityksiä käyttäen OTA-rajapintaa. Soveltuvia liiketoimialoja ovat mm. myyntiautomaatit, POS-kassajärjestelmät sekä kotisovellukset. /59/
- Wireless Module MC55/56 –moduulissa on dataliikenteen lisäksi myös ääniliikenne, joten kannettavat tietokoneet ja muut kannettavat päätelaitteet ovat sen sovellusalueita. /57/
- Wireless Module AC45 –moduulissa on lämpötilan tarkkailuun ja antennin diagnostiikkaan ohjelmistot. Moduuli toimii yli 75 °C lämpötilassa, joten se soveltuu auto-, kuljetus- ja logistiikkaratkaisuihin. /55/
- MC35i-päätelaitte on ensisijaisesti suunniteltu M2M-markkinoille. Sen pääasiallisia sovellusalueita ovat mittaus, etävalvonta, myyntiautomaatit, POS-kassajärjestelmät sekä kotisovellukset. /56/
- TC35i-päätelaitte on MC35i:n kaltainen, mutta siinä on SIM-kortin käsittelyä varten ohjelmisto sekä laajennetut AT-komennot teollisuussovelluksille. /58/

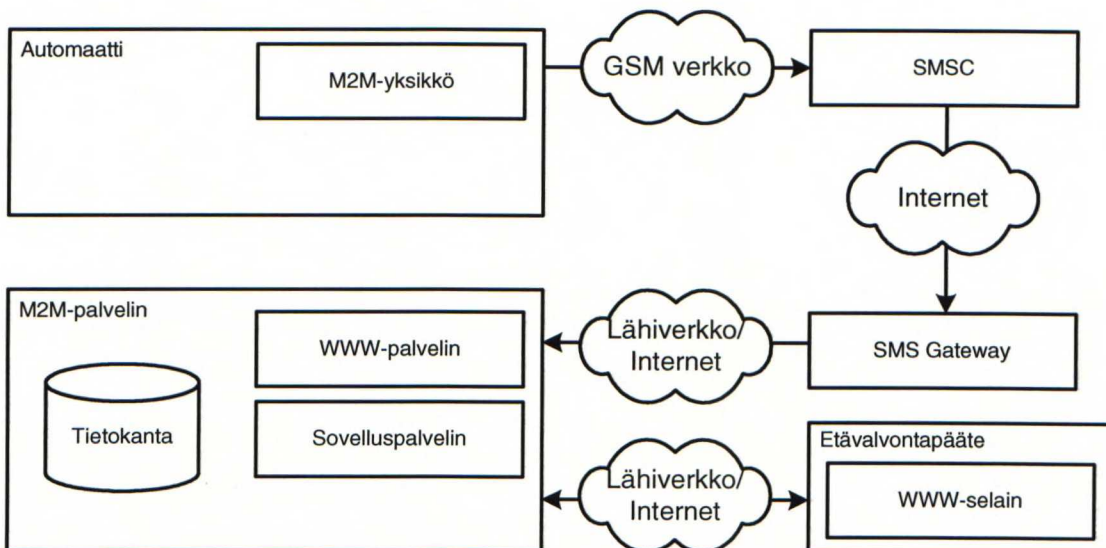
9 JÄRJESTELMÄN MÄÄRITELMÄ

9.1 Yleistä

Tässä luvussa esitetään järjestelmän arkkitehtuuri, eri rajapinnat, toiminnallisuus sekä tietomallin kuvaus. Koko rahapeliautomaattien langattomaan raportointijärjestelmään kuuluu seuraavat osat.

- Rahapeliautomaatti, josta raportoitava tieto kerätään
- Raportointiyksikkö, joka kerää ja lähettää raportoitavat tiedot
- Lyhytsanomakeskus, joka on GSM-verkon osa, jonka kautta viestit kulkevat
- SMS-yhdyskäytävä, jonka kautta viestit saadaan järjestelmään
- Viestien vastaanottopalvelu, joka vastaanottaa viestit
- Viestien tietovarasto, jonne tallennetaan saadut SMS-raportit
- Viestien käsittelypalvelu, joka jäsentää saadut SMS-viestit
- Järjestelmän tietovarasto, jossa automaattien tapahtuma- ja tilastotiedot sekä järjestelmän ja automaattien konfiguraatio sijaitsee
- Sovelluspalvelin, joka suorittaa järjestelmän vaatimat palvelut
- WWW-palvelin, jota tarvitaan mm. raportointiin

Kuvassa 11 on esitetty raportointijärjestelmän looginen arkkitehtuurikuvaus.



Kuva 11. Rahapeliautomaattien langaton raportointijärjestelmä

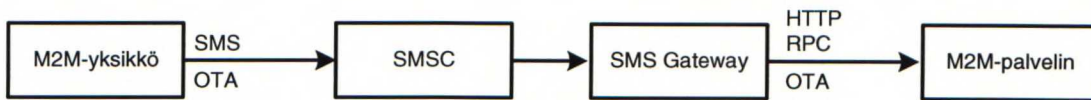
Järjestelmä ottaa automaattien raportointiyksikön lähettämiä viestejä vastaan. Viestit tulevat järjestelmään SMS-yhdyskäytävän kautta, jolloin ne tallennetaan

välivarastoon. Viestit ovat OTA-rajapinnan muodossa, joten ne täytyy jäsentää ennen tallentamista ja käsittelyä. Viestit tallennetaan relaatiotietokantaan, jossa on myös järjestelmän muut tiedot. Automaatilta saatujen tapahtuma- ja tilastotietojen perusteella luodaan tarvittaessa hälytyksiä ja raportteja.

9.2 Järjestelmän perusrakenne

9.2.1 Viestien vastaanotto SMS-yhdyskäytävästä

Automaatit lähettävät tapahtuma- ja tilastoviestejä raportointiyksikön kautta. Viestit ovat SMS-viestejä GSM-verkossa. Viestit tulevat SMSC-tekstiviestikeskuksen kautta SMS-yhdyskäytäväpalvelimelle (SMS Gateway). Yhdyskäytäväpalvelin määrittää välittämään saapuneet viestit ohjelmarajapintojen (API) kautta sovelluspalvelimelle. Sovelluspalvelimessa on palvelu, joka toteuttaa vaaditut ohjelmarajapinnat ja vastaanottaa viestit. Tämä palvelu on järjestelmän viestipalvelu.



Kuva 12. Viestien kulku valvontayksiköltä palvelimeen

SMS-yhdyskäytäväpalvelimesta riippuen ne toteuttavat mm. seuraavia ohjelmarajapintoja: HTTP, XML tai RPC. HTTP-rajapinta voidaan toteuttaa myös HTTP-protokollan salattua versiota eli HTTPS-protokollaa käyttäen. XML-ohjelmarajapinta voidaan toteuttaa HTTP-protokollan yli. Kuvassa 13 on esitettynä HTTP POST –metodilla lähetetyn viestin vastaanotto Java-servletissä. Servletti voi toimia itse viestipalveluna tai kutsua erillistä viestipalvelua, kuten esimerkiksi EJB-komponenttia.

```

public void doPost(
    HttpServletRequest request,
    HttpServletResponse response)
{
    String msisdn = request.getParameter("MSISDN");
    String message = request.getParameter("Message");

    RecordMessage(msisdn, message);
}
  
```

Kuva 13. Viestien vastaanotto Java-servletissä

RPC-ohjelmointirajapintaa käytettäessä luodaan RPC-kielellä tehdyn protokollan määritelmän mukaan palvelinrunko (server stub/skeleton) ja toteutetaan palvelinohjelma eli viestipalvelu. Kuvassa 14 on esitettynä RPC-protokollan määritelmä SMS-viestien vastaanottoon.

```
struct sms_message_struct
{
    char msisdn[16];
    string message <>;
};
typedef struct sms_message_struct sms_message_type;

program sms_service
{
    version sms_version
    {
        int receive_sms(sms_message_struct) = 1;
    } = 1;
} = 0x20000000;
```

Kuva 14. SMS-viestin RPC-protokollan määritelmä

RPC-palvelinrunko kutsuu toteutettavaa palvelinohjelmaa. Kuvassa 15 on kuvattu C-kielinen palvelinohjelma.

```
int * receive_sms_1(argp, rqstp)
    sms_message_struct *argp;
    struct svc_req *rqstp;
{
    static int result;

    result = record_message(argp->msisdn, argp->message);

    return (&result);
}
```

Kuva 15. SMS-viestin C-kielinen palvelinohjelma

9.2.2 Tietojen tallennus relaatiotietokantaan

SMS-yhdyskäytävältä saapuneista viesteistä tallennetaan tietokantaan lähettäjän MSISDN-numero sekä SMS-viestin runko. Lisäksi jokaiselle tapahtumalle luodaan yksikäsitteinen avain sekä kirjataan tapahtuman aikaleima, jotta voidaan tarkastella viestejä tulojärjestyksessä. SMS-viesti on OTA-rajapinnan määrittämässä muodossa, joten se vaatii jatkokäsittelyä ennen kuin siitä saadaan tarvittavat tiedot.

Java-kielessä voidaan tietokantayhteys luoda suoraan käyttäen DriverManager-luokkaa tai hakemalla se yhteysvarastosta (Connection pool). Kuvissa 16 ja 17 on kuvattuna tietokantayhteyden luominen molemmilla tavoilla.

```
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver").newInstance();
String url = "jdbc:odbc:m2mdb";
String username = "m2m";
String password = "m2m";
java.sql.Connection conn =
    DriverManager.getConnection(url, username, password);
```

Kuva 16. *Tietokantayhteys Java-kielellä DriverManager-luokan avulla*

```
javax.naming.Context ctx = new javax.naming.InitialContext();
javax.sql.DataSource ds = (javax.sql.DataSource)
    ctx.lookup("jdbc/ds_m2m");
java.sql.Connection conn = ds.getConnection();
```

Kuva 17. *Tietokantayhteys Java-kielellä JNDI-nimipalvelun avulla*

Kuvassa 18 on kuvattuna viestin tallennus tietokantaan Java-kielellä. Ennen viestin tallentamista on täytynyt luoda tietokannan yhteysolio, joka on esitetty nimellä conn.

```
Java.sql.PreparedStatement st = conn.prepareStatement(
    "INSERT INTO sms_receive (msisdn, message) VALUES (?, ?)");
st.setString(1, msisdn);
st.setString(2, message);
st.executeUpdate();
```

Kuva 18. *Viestin tallennus tietokantaan Java-kielellä*

Kuvassa 19 on esitettynä tietokantayhteyden luominen ja viestin tallennus tietokantaan Pro*C-kielellä. Esikäntäjä tekee Pro*C-ohjelmasta C-ohjelman, jolloin se voidaan liittää muuhun C-kieliseen toteutukseen.

```

EXEC SQL BEGIN DECLARE SECTION;
  VARCHAR conn_str[256];
  VARCHAR msisdn[24];
  VARCHAR message[24];
EXEC SQL END DECLARE SECTION;

void record_message(char* _msisdn, char* _message)
{
  strcpy((char *) msisdn.arr, _msisdn);
  msisdn.len = strlen((char *) msisdn.arr);

  strcpy((char *) message.arr, _message);
  message.len = strlen((char *) message.arr);

  conn_str.len = strlen(strcpy((char *) conn_str.arr,
    "m2m/m2m@m2mdb"));

  EXEC SQL CONNECT :conn_str;

  EXEC SQL INSERT INTO SMS_RECEIVE (MSISDN, MESSAGE)
    VALUES (:msisdn, :message);

  EXEC SQL COMMIT WORK RELEASE;
}

```

Kuva 19. Viestin tallennus tietokantaan Pro*C-kielellä

9.2.3 Viestien käsittely

Raportointiyksikön lähettämien SMS-viestien saavuttua järjestelmään täytyvät ne ensin jäsentää, jotta niistä voidaan muodostaa tapahtuma- ja tilastotietoja sovelluspalvelinta varten. Viestien käsittelypalvelu hakee saapuneet viestit, jäsentää ne ja tallentaa sen jälkeen tietokantaan.

Ensimmäiseksi viestistä käsitellään sen otsikko, joka on viestin alkuosassa. Otsikkotiedoista tarvitaan erityisesti viestintyyppi, joka kertoo onko kyseessä tapahtuma- vai tilastotieto. Lisäksi viestin pituudesta kertovaa tietoa voidaan käyttää eheyden tarkistamiseen. Viestinkäsittelyn pseudokoodi on esitetty kuvassa 20.

```

viesti ← lue_viesti();
otsikko ← hae_otsikko(viesti);
tyyppi ← hae_tyyppi(otsikko);
pituus ← hae_pituus(otsikko);
if(tyyppi = tapahtuma)
  Jäsennä tapahtuma viestistä;
if(tyyppi = tilasto)
  Jäsennä tilastot viestistä;

```

Kuva 20. Viestin käsittelyn pseudokoodi

Tapahtumaviestissä ei ole muuta tietoa kuin tapahtuman tunnistet ja mahdollisesti laskurin arvo. Tapahtumaviestin tiedot tallennetaan järjestelmän tietokantaan sille varattuun tietokantatauluun. Kuvassa 21 on tapahtumaviestin käsittelyn pseudokoodi.

```
tapahtuma ← hae_tapahtuma(viesti);
laskuri ← hae_laskurin_arvo(viesti);
tallenna_tapahtuma(msisdn, tapahtuma, laskuri);
```

Kuva 21. *Tapahtumaviestin käsittely*

Tilastoviestissä voi olla yhden tai useamman laskurin arvoja. Laskurikohtaisia arvoja voi olla myös useita, mutta niitä on yhtä monta jokaiselle laskurille. Tapahtumaviestissä on laskurikohtaisten arvojen määrä, ajankohdat jolloin tiedot on kerätty sekä laskurien arvot. Kuvassa 22 on tilastoviestin käsittelyn pseudokoodi.

```
arvoja ← hae_arvojen_määrä(viesti);
for(i←0; i<arvoja; i←i+1)
{
    päivä[i] ← hae_päivä(viesti, i);
    aika[i] ← hae_kellonaika(viesti, i);
}
while(viesti käsitelty kokonaan)
{
    laskuri ← hae_seuraava_laskurin_tunniste(viesti)
    for(j←0; j<arvoja; j←j+1)
    {
        arvo ← hae_laskurin_arvo(viesti, j);
        tallenna_arvo(msisdn, laskuri, arvo, päivä[j], aika[j]);
    }
}
```

Kuva 22. *Tilastoviestin käsittely*

9.2.4 Hälytykset

Järjestelmä muodostaa hälytyksiä automaattilta saatujen viestien perusteella. Viestien tiedot on tallennettu tietokantatauluihin, joten hälytykset muodostetaan tarkkailemalla näitä tauluja. Tarkkailu voidaan toteuttaa joko tietokannan laukaisimilla (database trigger) tai sovelluspohjaisesti.

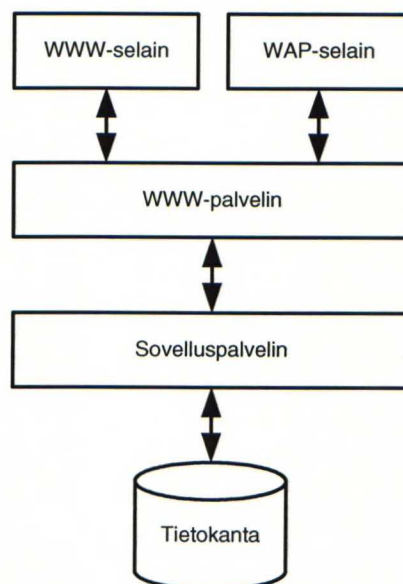
Kun tarkkailtavissa oleva muutos tapahtuu, tietokannan laukaisin kirjoittaa tiedon tapahtumasta automaattisesti tapahtumatauluun. Tapahtumataulusta voidaan käydä lukemassa luodut hälytykset ja välittää tieto eteenpäin.

Sovelluspohjainen ratkaisu perustuu tietyin väliajoin tehtyihin tietokantakyselyihin ja niiden vastausten perusteella tehtyihin päätelmiin. Sovelluspohjaisella hälytysmenetelmällä voidaan tehdä monipuolisempia hälytyssehtoja kuin laukaisimilla. Sovelluspohjainen menetelmä kuitenkin kuormittaa tietokantaa enemmän kuin laukaisimet, koska joudutaan tekemään kyselyitä tiedon etsimiseksi. Hälytyssovellus voi välittää tiedon suoraan eteenpäin tai laittaa sen tapahtumatauluun.

Hälytyspalvelu saa tiedon tapahtumataulusta tai hälytyssovellukselta ja muodostaa siitä viestin, joka välitetään SMS-yhdyskäytävälle. SMS-yhdyskäytävä lähettää viestin sille annettuun huoltomiehen MSISDN-numeroon.

9.2.5 Raportointi

Raporttien katselu tapahtuu etävalvontapääteeltä, jossa pääasiallisesti on WWW-selain. Selain hakee WWW-palvelimelta raportit, jotka ovat HTML- tai XHTML-sivuja. Mobiileita päätelaitteita varten, kuten WAP-selaimella varustettuja GSM-puhelimia, muodostetaan vastaavasti rajoitetummat WML-sivut. WWW-palvelin ottaa yhteyden sovelluspalvelimeen, jossa raportit muodostetaan. Sovelluspalvelin hakee tiedot raporteja varten tietokantapalvelimelta ja kokoaa tiedot raportiksi.



Kuva 23. Raportoinnin arkkitehtuuri

9.3 Tietokanta

9.3.1 Yleistä

Jokaisen automaatin tiedot tallennetaan tietokantaan. Automaatteja voi olla eri tyyppisiä, joten täytyy voida määritellä jokaiselle automaattityypille omat tapahtuma- ja tilastoviestit. Saapuvat SMS-viestit sekä niistä jäsennetyt viestit on tarve tallentaa jatkokäsittelyä ja raportointia varten. Tässä kappaleessa esitetään tarvittavien tietokantataulujen kuvaukset.

SMS-yhdyskäytävältä saapuvat ja lähtevät viestit ovat loogisesti erillään muista järjestelmän tietokantatauluista, mutta ne voidaan toteuttaa samaan tietokannan hallintajärjestelmään.

Tietokantataulujen kuvauksissa käytetään tietokannan hallintajärjestelmästä riippumattomia tietotyyppejä. Taulukossa 5 on kuvaus näistä tietotyypeistä.

Tietotyyppi	Kuvaus
N-Decimal(p, s)	Numeerinen arvo, missä p on koko luvun maksimi pituus ja s on desimaaliosa
C-Variable Length(n)	Vaihtuvan mittainen merkkijono, missä n on maksimi pituus
T-Date & Time	Päiväys ja kellonaika
R-Variable Length(n)	Binääristä tietoa, missä n on maksimi koko

Taulukko 5. Tietotyyppien kuvaukset

9.3.2 Automaatin tyyppi

Järjestelmä voi hallinnoida usean tyyppisiä automaatteja. Eri tyyppisistä automaateista raportoidaan mahdollisesti eri tietoja. Laitteiston tilasta kerättävät tiedot saattavat vaihdella automaatin rakenteesta riippuen. Esimerkiksi liikkuvat osat vaihtelevat automaattityypeittäin.

Automaattityypin yksikäsitteinen tunniste, nimi ja kuvaus ovat tietojen vähimmäisvaatimus. Taulun kuvaus on kuvassa 24.

MACHINE_TYPE	
MACHINE_TYPE_ID	N-Decimal(38,0)
NAME	C-Variable Length(50)
DESCRIPTION	C-Variable Length(200)

Kuva 24. MACHINE_TYPE-taulun kuvaus

9.3.3 Tyypikohtaiset tapahtumat

Automaatista saatavat tapahtumatiedot kuvataan tunnisteella. Tapahtumatiedon tunniste on koodi, jota vastaa selkokielen nimi. Tunniste voi olla numeerinen tai merkkimuotoinen koodi. Koodien määrä ja niiden merkitys saattavat vaihdella automaattityypeittäin. Taulun kuvaus on kuvassa 25.

EVENT	
MACHINE_TYPE_ID (FK)	N-Decimal(38,0)
EVENT_CODE	C-Variable Length(10)
NAME	C-Variable Length(50)

Kuva 25. EVENT-taulun kuvaus

9.3.4 Tyypikohtainen tilastotieto

Automaatista saatavat tilastotiedot kuvataan tunnisteella. Tilastotiedon tunniste vastaa automaatissa olevia laskureita. Laskureiden määrä ja niiden merkitys saattavat vaihdella automaattityypeittäin. Laskureiden tunniste on koodi, jota vastaa selkokielen nimi. Tunniste voi olla numeerinen tai merkkimuotoinen koodi. Taulun kuvaus on kuvassa 26.

STATUS	
MACHINE_TYPE_ID (FK)	N-Decimal(38,0)
STATUS_CODE	C-Variable Length(10)
NAME	C-Variable Length(50)

Kuva 26. STATUS-taulun kuvaus

9.3.5 Automaatti

Automaatin tärkein tieto on sen raportointiyksikössä olevan SIM-kortin tilaajanumero eli MSISDN. Tämä riittää yksilöimään automaatin. Jotta järjestelmään voitaisiin lisätä automaatteja ennen kuin tiedetään sen MSISDN-numero, tarvitsee se yksikäsitteisen tunnusteen. Automaatti kuuluu myös johonkin kategoriaan, joka määrää automaattien

yhteiset ominaisuudet kuten raportoitavat tapahtumatiedot. Lisäksi automaatile voidaan antaa nimi, paikka ja lisätietoja. Taulun kuvaus on kuvassa 27.

MACHINE

MACHINE_ID	N-Decimal(38,0)
MACHINE_TYPE_ID (FK)	N-Decimal(38,0)
MSISDN	N-Decimal(15,0)
NAME	C-Variable Length(50)
LOCATION	C-Variable Length(50)
DESCRIPTION	C-Variable Length(50)

Kuva 27. MACHINE-taulun kuvaus

9.3.6 Automaatin tapahtumat

Automaatista tulevasta tapahtumatiedosta tallennetaan tapahtumaa kuvaava koodi. Jos viestissä on ollut myös laskurin arvo, tallennetaan myös se. Lisäksi tallennetaan tieto viestin vastaanottamisen ajasta. Taulun kuvaus on kuvassa 28.

MACHINE_EVENT

MACHINE_EVENT_ID	N-Decimal(38,0)
MACHINE_ID (FK)	N-Decimal(38,0)
EVENT_CODE	C-Variable Length(10)
VALUE	C-Variable Length(10)
RECEIVE_TIME	T-Date & Time

Kuva 28. MACHINE_EVENT-taulun kuvaus

9.3.7 Automaatin tilastotieto

Automaatista tulevasta tilastotiedosta tallennetaan laskuria kuvaava tunniste, laskurin arvo sekä tiedon keräyspäivää ja aikaa kuvaavat tiedot. Lisäksi tallennetaan tieto viestin vastaanottamisen ajasta. Taulun kuvaus on kuvassa 29.

MACHINE_STATUS

MACHINE_STATUS_ID	N-Decimal(38,0)
MACHINE_ID (FK)	N-Decimal(38,0)
STATUS_CODE	C-Variable Length(10)
COUNTER	N-Decimal(38,0)
REPORT_TIME	C-Variable Length(10)
RECEIVE_TIME	T-Date & Time

Kuva 29. MACHINE_STATUS-taulun kuvaus

9.3.8 Saapuneet viestit

Saapuvasta SMS-viestistä tallennetaan lähettäjän MSISDN-numero, itse viesti ja aikaleima tapahtumasta. Lisäksi sillä on yksikäsitteinen tunniste. Taulun kuvaus on kuvassa 30.

RECEIVED_SMS	
MESSAGE_ID	N-Decimal(38,0)
MSISDN	N-Decimal(15,0)
MESSAGE	C-Variable Length(160)
RECEIVE_TIME	T-Date & Time

Kuva 30. RECEIVED_SMS-taulun kuvaus

9.3.9 Lähtevät viestit

Lähetettävästä SMS-viestistä tallennetaan vastaanottajan MSISDN-numero, itse viesti, viestin tila, luonti- sekä lähetysaika. Lisäksi sillä on yksikäsitteinen tunniste. Viestin tila kertoo onko se lähetetty tai onko lähetys epäonnistunut. Luontiajankohta on aikaleima siitä, koska rivi on luotu tietokantaan. Lähetysaika kertoo sen, koska viesti on saatu toimitettua SMS-yhdyskäytävälle. Taulun kuvaus on kuvassa 31.

SEND_SMS	
MESSAGE_ID	N-Decimal(38,0)
MSISDN	N-Decimal(15,0)
MESSAGE	C-Variable Length(160)
STATUS	N-Decimal(1,0)
CREATION_TIME	T-Date & Time
SEND_TIME	T-Date & Time

Kuva 31. SEND_SMS-taulun kuvaus

9.3.10 Järjestelmän käyttäjät

Järjestelmän käyttäjistä tallennetaan nimi, käyttäjätunnus, salasana sekä rooli. Lisäksi käyttäjillä on yksikäsitteinen tunniste järjestelmässä. Käyttäjän rooli kertoo mitä oikeuksia järjestelmän käyttöön on. Erilaisia oikeuksia ovat mm. raporttien katselu, käyttäjien lisäys ja järjestelmän tietojen muuttaminen. Salasana tallennetaan yksisuuntaisella salausalgoritmillä, joten sitä ei voi selvittää tietokannan tiedosta. Sopiva algoritmi tähän on esimerkiksi MD5 /50/. Taulun kuvaus on kuvassa 32.

USERS

USER_ID	N-Decimal(38,0)
NAME	C-Variable Length(30)
USERNAME	C-Variable Length(10)
PASSWORD	R-Variable Length(16)
ROLE	N-Decimal(38,0)

Kuva 32. *USERS-taulun kuvaus*

9.3.11 Hälytykset

Automaatin raportointiyksikön välittämien tapahtumaviestien pohjalta voidaan lähettää hälytysviestejä. Hälytystaulussa on lueteltuna automaatin tunnistetapahtumatyyppi sekä hälytyksen vastaanottaja. Taulun kuvaus on kuvassa 33.

ALERTS

ALERT_ID	N-Decimal(38,0)
MACHINE_ID (FK)	N-Decimal(38,0)
ALERT_CODE	C-Variable Length(10)
RECEIVER_MSISDN	N-Decimal(15,0)

Kuva 33. *ALERTS-taulun kuvaus*

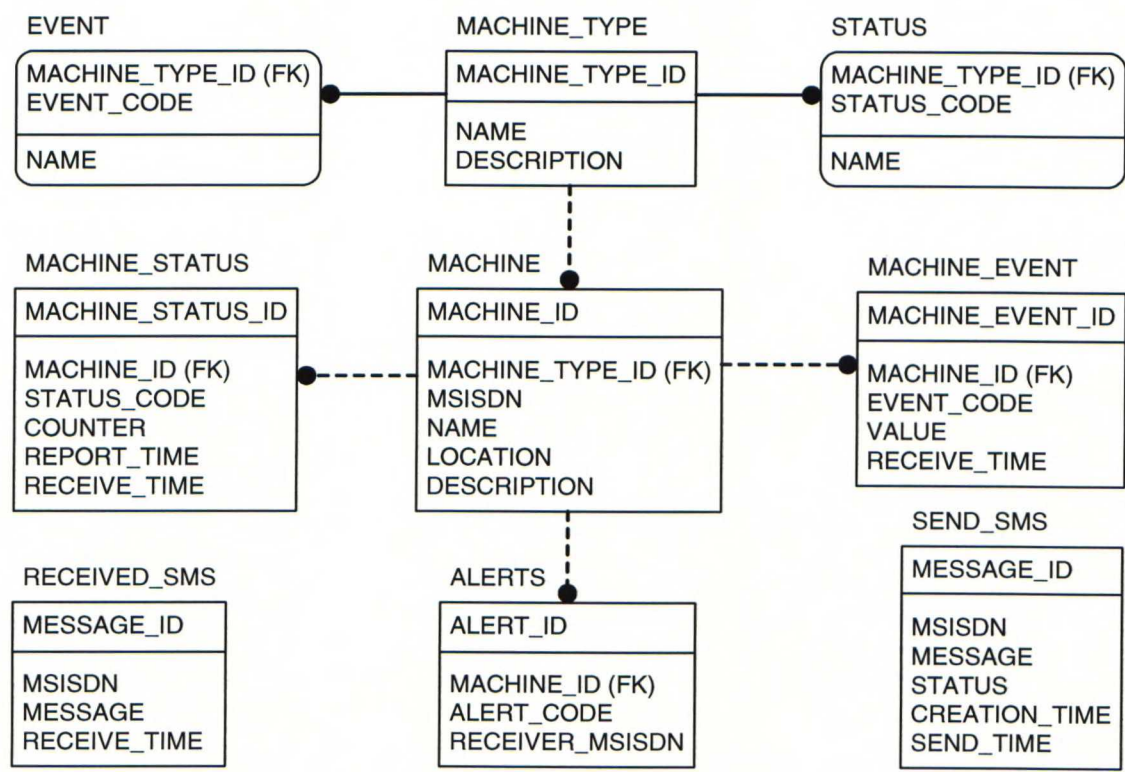
9.3.12 Tietokantamalli

Kuvassa 34 on esitetty tietokantamalli, jossa on viestinvälitykseen ja automaatteihin liittyvät tietokantataulut. Malli on esitetty IDEF1X-tiedonkuvausmenetelmällä. Taulukossa 6 on esitetty käytettyjen symbolien merkitys. /39/

Saapuneilla ja lähtevillä viesteillä ei ole suoraa suhdetta automaattiin, vaikka ne yhdistyvät toisiinsa MSISDN-numeron avulla. Lähteviin viesteihin tulee myös viestit, jotka lähetetään huoltomiehille. Saapuneisiin viesteihin tulee kaikki vastaanotetut viestit riippumatta siitä onko kyseisellä MSISDN-numerolla olevaa automaattia kirjattu järjestelmään. Tämä mahdollistaa viestien yhdistämisen tiettyyn automaattiin myös myöhemmin. Automaatin tapahtuma- ja tilastoviestillä ei ole suoraa suhdetta tapahtuma- ja tilastoviestien kuvauksiin, mikä mahdollistaa kaikkien automaatilta saapuneiden viestien kirjaamisen riippumatta siitä onko sille kuvausta.

Merkintä	Kuvaus
Entiteetin nimi Avainattribuutit Muut attribuutit	Riippumaton entiteettityyppi on sellainen, minkä instanssi voi olla olemassa ilman muita
Entiteetin nimi Riippuvat avainattribuutit (FK) Muut avainattribuutit Muut attribuutit	Riippuvainen entiteettityyppi on sellainen, minkä instanssin olemassa olo on riippuvainen muista instansseista. Avainattribuutit, joista on viittaus muualle, merkitään FK-tunnuksella (foreign key)
————●	Yksilöivä suhde
-----●	Yksilöimätön suhde

Taulukko 6. Tietokantamallin merkintöjä /39/



Kuva 34. Tietokantamalli

9.4 Järjestelmän käyttö ja ylläpito

9.4.1 Yleinen käyttö

Järjestelmää voidaan käyttää WWW- tai WAP-selaimella. Järjestelmään kirjaudutaan sisään käyttäjätunnuksella ja salasanalla. Jokaisella käyttäjällä on oma tunnuksensa, mutta voidaan käyttää myös yhteiskäyttöisiä tunnuksia.

Järjestelmästä voidaan hakea tapahtuma- ja tilastotietoja aikajaksoittain tai peliautomaattikohtaisesti. Lisäksi voidaan hakea käyttäjälle suunnattuja hälytyksiä.

Järjestelmän kautta voidaan myös lähettää automaateille kyselyitä. Kyselyiden lähettäminen on tarpeellista, jos halutaan sellaista tietoa mitä ei tule normaaleiden tilasto- ja tapahtumaviestien mukana. Jos huomataan ettei jotain tiettyä tilastoviestiä ole saapunut automaatilta järjestelmään, voidaan pyytää näiden tietojen uudelleen lähetystä.

9.4.2 Ylläpito

Järjestelmän ylläpito tapahtuu WWW-sivujen kautta. Tietoliikenneyhteys salataan ylläpitosivuille HTTPS-protokollaa käyttäen. Jotta ylläpitoa voidaan käyttää, tarvitaan käyttäjätunnus ja salasana. Käyttäjien tunnukset ja käyttöoikeudet on määritelty USERS-tietokantataulussa. Ylläpidossa voidaan katsoa, lisätä ja muuttaa käyttäjien, viestityyppien sekä järjestelmässä olevien automaattien tietoja. Käyttäjän käyttöoikeustaso voi rajata oikeuden esimerkiksi pelkästään tiedon katseluun.

Ylläpidossa voidaan lisätä järjestelmään uusia peliautomaatteja ja automaattiryhmiä sekä määritellä raportoitavat tapahtuma- ja tilastotyyppit. Jokaisella automaattiryhmällä on omat tapahtuma- ja tilastotyyppit. Tapahtuma- ja tilastotyyppiin määritellään automaateilta tuleva laskurin koodi sekä sitä vastaava selite. Esimerkiksi koodi '45' voi tarkoittaa huolto-oven aukaisua.

9.4.3 Käyttö WAP-selaimella

WAP-selainten koko rajoittaa suuresti niiden käyttöä. Tästä johtuen WAP-selainten käyttö voidaan rajata vain viimeisimmän tiedon hakuun. Kuvassa 35 on esitetty esimerkki WAP-selaimen käytöstä, kun haetaan peliautomaatin tapahtumatietoja.

Näyttöjen ulkoasu voi vaihdella selaimen ja päätelaitteen tyyppin mukaan. Aloitus sivulla on sisäänkirjautumista varten kaksi kenttää, joihin tulee käyttäjätunnus sekä salasana. Lisäksi sivulla on linkki sisäänkirjautumiselle sekä kenttien tyhjennykselle. Päävalikosta valitaan haettavan tiedon tyyppi ja alivalikossa tarkennetaan hakua. Seuraavaksi valitaan valikosta haluttu peliautomaatti ja etsitään sen viimeisimmät tiedot. Raportissa näkyy tapahtumien nimet sekä niitä vastaavat laskureiden arvot ja aikaleimat lyhyessä muodossa.

The figure displays five sequential screens of the Wireless Reporting System interface:

- 1. Aloitus sivu (Login):** Features a 'Wireless Reporting System' header, 'Username' and 'Password' input fields, and 'Login' and 'Reset' buttons.
- 2. Päävalikko (Main Menu):** Features a 'Wireless Reporting System' header, a 'Main Menu' section, and links for 'Alerts', 'Event reports', and 'Status reports'.
- 3. Tapahtumaraaporttien valikko (Event reports):** Features a 'Wireless Reporting System' header, an 'Event reports' section, and links for 'Machine events' and 'Last events'.
- 4. Peliautomaatin valinta (Machine events):** Features a 'Wireless Reporting System' header, a 'Machine events:' section, a search box containing 'BACTA loc. 1234', and 'Find' and 'Back' buttons.
- 5. Viimeisimmät tapahtumat (Latest events):** Features a 'Wireless Reporting System' header, a 'BACTA loc. 1234:' section, a table of events, and 'Next' and 'Back' buttons.

Event	Value	Time	Date
HW Error	222	20:11	14.12
Service			
Door	5212	20:09	14.12
Open			

Kuva 35. WAP-selaimen käyttö

9.5 Raportit

9.5.1 Yleistä

Järjestelmä tuottaa raportteja WWW- ja WAP-selaimille. WAP-selaimille muodostetut raportit ovat suppeampia, joten ne sisältävät vain perustiedot halutusta raportista.

Raportit muodostetaan hakuetojen perusteella. Hakuetoja ovat tyyppi, aikaväli, tapahtumien tyypit ja arvot sekä raportoitavat peliautomaatit. Raportin tyyppi on tapahtuma- tai tilastotieto. Aikavälillä voidaan määrittää raportit päivä-, viikko-, kuukausi- tai vuositasolla. Tapahtumien tyyppillä ja arvolla voidaan rajoittaa

näytettävän tiedon määrä. Raportit voidaan myös muodostaa peliautomaatti- tai ryhmäkohtaisiksi. Lisäksi voidaan muodostaa raportteja käyttäjille luoduista hälytyksistä.

9.5.2 Tilastoraportit

Tilastoraporteissa esitetään peliautomaateilta säännöllisesti saatuja tilastotietoja, jotka ovat automaatin laskureiden arvoja. Tilastotiedoista voidaan tarkastella viimeisintä tietoa tai niiden kehitystä jollain aikajaksolla. Kuvissa 36 ja 37 on esitetty esimerkkejä tilastoraporteista. Tilastoraportti 1 sisältää laskureiden kokonaisarvot eri ajankohtina. Tilastoraportissa 2 esitetään laskureihin tullutta muutosta viikkotasolla. Tilastoraportit voivat olla joko automaattikohtaisia tai kooste automaattiryhmän tiedoista.

STATUS	VALUE	TIMESTAMP
Coin in 20p	543210	22:05:10 9.12.2004
Coin in 50p	32109	22:05:10 9.12.2004
Coin in £1	432109	22:05:10 9.12.2004
Coin in 20p	543077	22:07:29 8.12.2004
Coin in 50p	32002	22:07:29 8.12.2004
Coin in £1	432059	22:07:29 8.12.2004

Kuva 36. Tilastoraportti 1

STATUS	WEEK 41	WEEK 42	WEEK 43
Coin in 20p	4321	5432	6543
Coin in 50p	123	0	2109
Coin in £1	345	395	390

Kuva 37. Tilastoraportti 2

9.5.3 Tapahtumaraportit

Tapahtumaraporteissa esitetään tietoja automaateista, mitä ei raportoida säännöllisesti. Tällaisia tapahtumia ja tietoja automaatista ovat mm. rahaoven avaus, laitteistovika, huolto-oven avaus, laskurin vikaantuminen tai ilmoitus rahantäytön tarpeellisuudesta. Tapahtumaraporteissa näytetään tapahtuman nimi sekä sen ajankohta. Lisäksi raporteissa voi olla tapahtumien yhteismäärä samaan tapaan kuin tilastoraporttien laskureiden arvo. Tapahtumaraportit voidaan koostaa tietyn automaatin osalta tai tarkastella automaattiryhmää.

9.5.4 Hälytysraportit

Hälytysraportit muodostetaan automaateilta saatujen tapahtumatiетоjen pohjalta. Hälytys voi muodostua yksittäisestä tapahtumasta tai sarjasta tapahtumia. Yhteen hälytykseen liittyvien tapahtumien ei tarvitse olla samoja. Hälytys voi muodostua esimerkiksi siitä, jos vuorokauden sisällä tulee enemmän kuin kolme ilmoitusta automaatin laitteisto- tai muistiviasta. Hälytysraportit sisältävät myös tiedon siitä, onko huoltomiehelle lähetetty SMS-viestillä tieto hälytyksestä.

10 PILOTTIPROJEKTI

10.1 Yleistä

Tässä luvussa tarkastellaan langattoman raportointijärjestelmän toimivaa prototyyppiä, jossa testattiin järjestelmän komponenttien integraation toimivuutta. Projektissa kerättiin tilasto- ja tapahtumatietoja kahdesta rahapeliautomaatista, jotka oli sijoitettu eri puolille Eurooppaa. Pilottiprojektissa ei ollut tarkoitus testata suorituskkyä tai toteuttaa laajamittaista sovelluksen loogista toiminnallisuutta eli business-logiikkaa sovelluspalvelimelle.

10.2 Järjestelmän komponentit

10.2.1 Pelikoneet

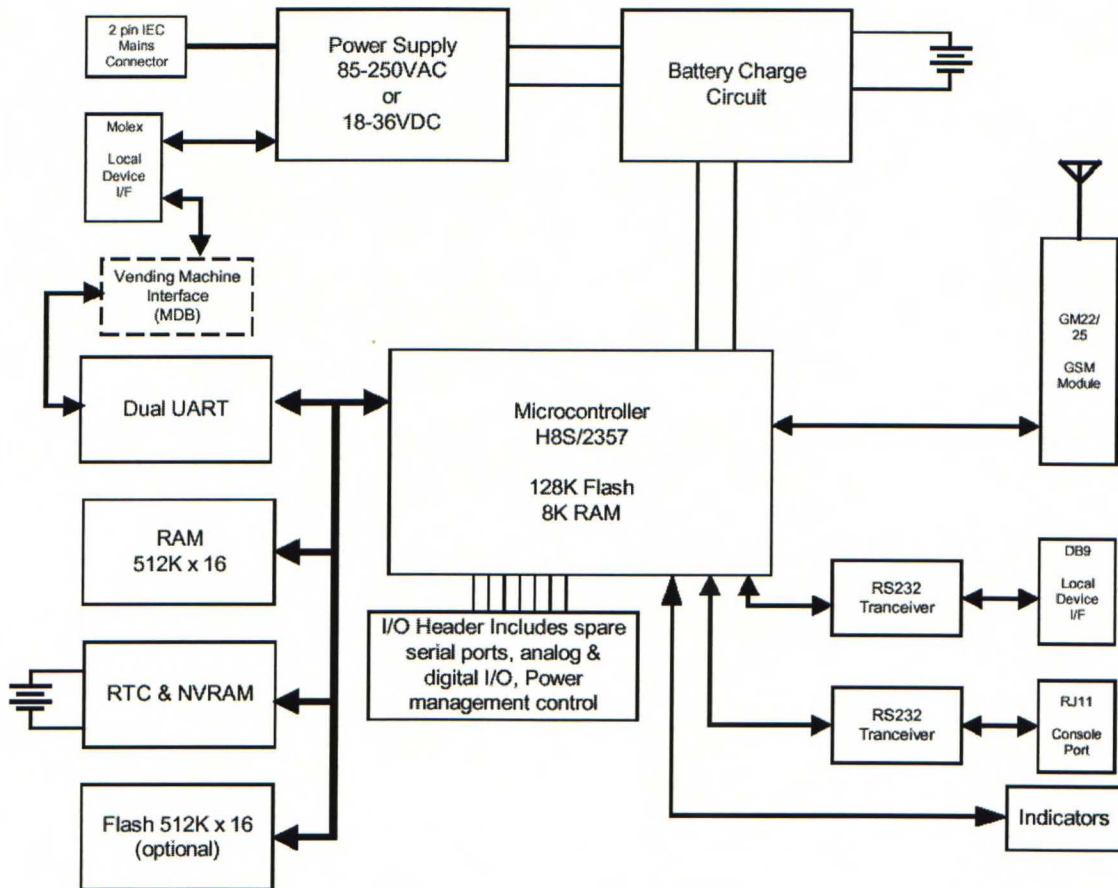
Pilottiprojekti koostui kahdesta pelikoneesta, jotka olivat varustettu langattomilla raportointiyksiköillä. Toinen automaateista oli Barcrestin /7/ ja toinen Cirsan /10/ pelikone. Barcrestin The Addams Family rahapeliautomaatti oli sijoitettu toimistolle Espooseen ja Cirsan Corsarios pelikone oli sijoitettu Espanjassa Málagan Fuengirolassa sijaitsevaan Yorkshire Tavern ravintolaan.

Lisäksi käytettiin kolmea matkapuhelinta virtuaalisina peliautomaatteina. Matkapuhelimista voitiin lähettää oikeassa muodossa olevia tilasto- ja tapahtumaviestejä tekstiviestinä palvelimelle.

10.2.2 Raportointiyksiköt

Rahapeliautomaattien raportointiyksiköt koostui alihankkijan suunnittelemaasta ja toteuttamasta alustasta. Kuvassa 38 on esitettynä raportointiyksikön laitteistoarkkitehtuuri. Alustan keskeisimmät osat ovat Hitachin 16-bittinen H8S/2357-mikrokontrolleri, Ericssonin GM 25 GSM-modeemi, RAM-muistipiiri, virtalähde akkuvarmennuksella, reaaliaikakellopiiri haihtumattomalla muistilla sekä viisi sarjaporttia. 9-napaista RS-232 ja kahta Multi-Drop Bus -liitäntää käytetään yksikön liittämiseen rahapeliautomaattiin. RJ-11 liitäntä on konsolia varten, mitä voidaan käyttää monitorointiin tai ohjelmiston päivitykseen. Lisäksi on liitäntä GSM-modeemia varten. Käyttöjärjestelmänä raportointiyksikössä on μ C/OS-II eli

MicroController Operating System Version 2. Toiminnallisuus ohjelmoitiin raportointiyksikköön C-kielellä sekä symbolisella konekielellä. /52, 54, 68/



Kuva 38. Raportointiyksikön laitteistoarkkitehtuuri /68/

10.2.3 Palvelimet

Käytössä oli kaksi Sunin SPARC/Solaris-palvelinta. Toiseen palvelimeen oli asennettu SMS-yhdyskäytäväksi Exomi SMS Gateway /19/. Toiseen palvelimeen oli asennettu tietokantapalvelimeksi Oracle /46/, WWW-palvelimeksi Apache HTTP Server /5/ sekä PHP-kielen tulkki /48/.

10.2.4 Lyhytsanomapalvelut

Lyhytsanomaviestipalvelut ja -numerot tilattiin Soneralta. Peliautomaattien raportointiyksiköiden liittymät olivat myös samalta teleoperaattorilta. Raportointiyksiköt määriteltiin lähettämään tapahtuma- ja tilastoviestit saatuun lyhytsanomaviestinumeroon. Automaattien lähettämät tekstiviestit tulivat

teleoperaattorin lyhytsanomakeskuksen kautta palvelimella olevaan SMS-yhdyskäytävään. Yhdyskäytävän kautta voitiin myös lähettää kyselyitä automaateille.

10.2.5 Sovellukset

Palvelimelle tehtiin RPC-palvelu, jota SMS-yhdyskäytävä kutsui saatuaan viestin peliautomaatilta. RPC-palvelu jäsensi ja tallensi viestin sisällön tietokantaan.

Viestien lähettämiseen automaateille käytettiin SMS-yhdyskäytävän asiakassovellusta. Asiakassovellus oli komentorivipohjainen ohjelma ja sen toiminta perustui RPC-protokollaan. Yhdyskäytävän asiakassovellusta kutsuttiin WWW-sivujen kautta PHP-kielen komennolla.

10.2.6 Käyttöliittymä ja raportit

Pilottiprojektiin tehtiin käyttöliittymä WWW-selaimille. Tiedonhaku ja logiikka tehtiin sivuille PHP-kielellä. Sivuston kautta pystyttiin määrittelemään uusia automaatteja sekä tilasto- ja tapahtumatyyppejä, lähettämään pyyntöjä automaateille sekä katsomaan automaattien tilasto- ja tapahtumaraportteja.

Kuvassa 39 on esitetty automaatin tilastoraportti. Vasemmalla on käyttöliittymän valikko ja oikealla itse raportti. Raportissa näkyy kahden päivän tilastotietoja. Raportissa on laskureiden selkokielineen nimi, niitten arvo sekä tietojen vastaanottoaika. Ylimmällä rivillä on laskurin 350 arvo, mutta sille ei ole annettu selkokielistä kuvausta, kuten muille laskureille, joita raportissa esiintyy.

Reports STATUS ALARM	MACHINE STATUS: Yorkshire Tavern, Fuengirola, Spain (Cirsa Corsarios)		
	SLOT	VALUE	RECEIVE TIME
Send MESSAGE	350	1	13:25:21 13.11.2001
	Coin in 25 ptas	23885	13:25:20 13.11.2001
Configure MACHINE ALARM STATUS MACHINE TYPE	Money out (25 ptas units)	478000	13:25:20 13.11.2001
	Money to cashbox (25 ptas units)	180641	13:25:20 13.11.2001
	100 ptas coin to cashbox	25983	13:25:20 13.11.2001
	25 ptas coin to cashbox	6529	13:25:20 13.11.2001
	Coin in 100 ptas	149739	13:25:20 13.11.2001
	Coin in 200 ptas	494	13:25:20 13.11.2001
	Coin in 50 ptas	284	13:25:20 13.11.2001
	Coin in 500 ptas	3283	13:25:20 13.11.2001
	Coin out 100 ptas	123100	13:25:20 13.11.2001
	Coin out 25 ptas	14040	13:25:20 13.11.2001
	Money in (25 ptas units)	664585	13:25:20 13.11.2001
	Coin in 25 ptas	23859	15:01:54 12.11.2001
	Money out (25 ptas units)	476580	15:01:54 12.11.2001
	Money to cashbox (25 ptas units)	179997	15:01:54 12.11.2001
	100 ptas coin to cashbox	25856	15:01:54 12.11.2001
	25 ptas coin to cashbox	6529	15:01:54 12.11.2001

Kuva 39. Automaatin tilastoraportti

10.3 Pilottiprojektin johtopäätökset

Rakennettu prototyyppi langattomasta raportointijärjestelmästä toimi sille odotetulla tavalla ja täytti sille asetetut tavoitteet. Tarkoituksena oli osoittaa järjestelmän integraation toimivuus. Toimivuus pystyttiin selkeimmin osoittamaan luomalla raportteja peliautomaateilta tulleista tiedoista. Raportit olivat luonteeltaan tilasto- ja tapahtumatietoja. Tilastotietoja olivat esimerkiksi eri rahalaskureiden arvot tiettyinä päivinä. Tapahtumatietoja tai hälytyksiä puolestaan olivat tiedot mm. rahaoven avauksista.

11 JOHTOPÄÄTÖKSET

11.1 Yhteenveto

Työssä tarkasteltiin rahapeliautomaattien liiketoimintaympäristöä ja menetelmiä tapahtumapohjaisissa järjestelmissä. Tapahtumapohjaisiin järjestelmiin kuuluvat kommunikaatiojärjestelmät sekä valvontajärjestelmät. Lisäksi selvitettiin langattoman raportointijärjestelmän kannalta oleellisia järjestelmiä kuten tietoliikenne GSM-järjestelmässä sekä tietokantajärjestelmä.

Työssä määriteltiin rahapeliautomaattien langattoman raportointijärjestelmän arkkitehtuuri. Tämä sisälsi mm. käytettävien tiedonsiirtoprotokollien, kommunikointitapojen, ohjelma- ja tietokantarajapintojen, palvelinohjelmiston tarkastelua sekä relaatiotietokannan kuvauksen. Keskeisimpiä kommunikointitapoja järjestelmässä ovat HTTP, RPC, lyhytsanomat sekä OTA-viestit. Tiedon esitys tapahtuu HTML- ja WML-sivujen avulla. Oleellisimmat tietokantarajapinnat ovat ODBC ja JDBC. Järjestelmän kannalta tärkeät palvelimet ovat relaatiotietokannan hallintajärjestelmä, sovelluspalvelin, SMS-yhdyskäytävä sekä WWW-palvelin.

Lopuksi kuvattiin pilottiprojekti, jossa testattiin järjestelmän komponenttien integraatiota. Projektissa oli kaksi pelikonetta raportointiyksiköineen eri puolilla Eurooppaa. Lisäksi käytettiin matkapuhelimia virtuaalisina pelikoneina. Pelikoneilta saaduista tapahtuma- ja tilastotiedoista muodostettiin HTML-raportteja.

11.2 Tavoitteiden täyttyminen

11.2.1 Yleistä

Rahapeliliiketoiminnan asettamat vaatimukset langattomalle raportointijärjestelmälle liittyivät reaaliaikaisuuteen, joustavuuteen sekä luotettavuuteen. Tässä kappaleessa esitetään näiden vaatimuksien täyttyminen työssä määritellyn arkkitehtuurin osalta.

11.2.2 Reaaliaikaiset hälytykset

Automaattien tapahtumatiedot välitetään järjestelmään lyhytsanomien eli tekstiviestien välityksellä. Saapuneiden viestien perusteella voidaan luoda

hälytysviestejä ennalta määrättyyn numeroon. Tekstiviestien viiveet ovat yleisesti ottaen lyhyitä, joten tämä täyttää määritellyn vaatimuksen reaaliaikaisista hälytyksistä. GSM-verkko ei kuitenkaan takaa lyhytsanomien perillemeno.

11.2.3 Säännöllinen tiedonkeruu

Automaateista voidaan vastaanottaa viestejä säännöllisin väliajoin. Tietokantamalli tukee raportointivälin vaihtelua ja raportointi voi tapahtua kollektiivisesti jälkeinpäin.

11.2.4 Joustava ympäristö

Järjestelmään voidaan liittää uusia automaatteja järjestelmän käyttöönoton jälkeen. Automaatit on lajiteltu tyypeittäin ja raportoitavat tilasto- ja tapahtumatiedot voidaan määritellä tyyppikohtaisesti. Näin ollen viestejä voidaan vastaanottaa erilaisilta automaateilta. Käytettävien tietokantarajapintojen avulla järjestelmä voidaan liittää suoraan myös muihin järjestelmiin.

11.2.5 Kaksisuuntainen tietoliikenne

Järjestelmään määritelty SMS-yhdyskäytävä mahdollistaa tietoliikenteen myös automaatin suuntaan. Järjestelmän määritys ei aseta esteitä kaksisuuntaiselle tietoliikenteelle, mutta automaateille lähetettävien viestien määrittely on rajattu tämän työn ulkopuolelle. Hälytysviestien lähettäminen huoltomiehille on kuitenkin määritelty.

11.2.6 Luotettavuus

Järjestelmä mahdollistaa kaksisuuntaisen tietoliikenteen, joten saapumattomien tilastoviestien pyytäminen automaatilta jälkeinpäin on mahdollista. Tilastotietoihin tallennetaan aina automaatin lähettämien nollaamattomien laskuritietojen arvot. Näin ollen yksittäisten tietojen puuttuminen järjestelmästä ei muuta tiedon oikeellisuutta.

11.2.7 Langattomat päätelaitteet

Automaateilta kerättyä tietoa voidaan tarkastella WML-dokumentteina WAP-selaimella varustetuilla päätelaitteilla. Järjestelmän hälytykset lähtevät lyhytsanomaviesteinä määriteltyyn matkapuhelinnumeroon. Näin ollen järjestelmä tukee langattomia päätelaitteita tiedon saannissa.

12 LÄHDELUETTELO

- /1/ **ANSI X.135.** *Database Language SQL*. American National Standards Institute, 1986.
- /2/ **ANSI X.135.** *Database Language SQL with Integrity Enhancement*. American National Standards Institute, 1989.
- /3/ **ANSI X.135.** *Database Language SQL*. American National Standards Institute, 1992.
- /4/ **ANSI X.135.** *Database Language SQL*. American National Standards Institute, 1999.
- /5/ **Apache.** *Apache HTTP Server Project*. 2004 (online) [viitattu 1.12.2004]. Saatavilla WWW-muodossa: <URL:<http://httpd.apache.org/>>
- /6/ **BACTA,** *British Amusement Catering Trades Association*. 2004 (online) [viitattu 1.12.2004]. Saatavilla WWW-muodossa: <URL:<http://www.bacta.org.uk>>.
- /7/ **Barcrest Games.** *Barcrest Games*. N.d. (online) [viitattu 1.12.2004]. Saatavilla WWW-muodossa: <URL:<http://www.barcrest.co.uk>>
- /8/ **BEA.** *BEA WebLogic Server*. 2004 (online) [viitattu 1.12.2004]. Saatavilla WWW-muodossa: <URL:<http://www.bea.com/framework.jsp?CNT=index.htm&FP=/content/products/server/>>
- /9/ **Bloomer, J.** *Power Programming with RPC*. O'Reilly, 1992.
- /10/ **CIRSA.** *CIRSA*. 2004 (online) [viitattu 1.12.2004]. Saatavilla WWW-muodossa: <URL:<http://www.cirsa.com/>>
- /11/ **Codd, E.** *A Relational Model of Data for Large Shared Data Banks*. Communications of the ACM, 1970.
- /12/ **Crispin, M.** *INTERNET MESSAGE ACCESS PROTOCOL – VERSION 4rev1*. Request For Comments 3501, 2003.
- /13/ **DeMichiel, L.** *Enterprise JavaBeans™ Specification, Version 2.1*. Sun Microsystems 2003.
- /14/ **Dierks, T., Allen, C.** *The TLS Protocol Version 1.0*. Request For Comments 2246, 1999.
- /15/ **Ellis, J., Ho, L.** *JDBC™ 3.0 Specification*. Sun Microsystems, 2001.
- /16/ **ETSI GSM 03.40.** *Digital cellular telecommunications system (Phase 2+); Technical realization of the Short Message Service (SMS); Point-to-Point (PP)*. European Telecommunications Standards Institute, 1998.
- /17/ **ETSI GSM 02.16.** *Digital cellular telecommunications system (Phase 2+); International Mobile station Equipment Identities (IMEI)*. European Telecommunications Standards Institute, 1997.
- /18/ **ETSI GSM 03.38.** *Digital cellular telecommunications system (Phase 2+); Alphabets and language-specific information*. European Telecommunications Standards Institute. 1998

- /19/ **Exomi.** *SMS Gateway Module of EMG*. 2004 (online) [viitattu 1.12.2004].
Saataavilla WWW-muodossa:
<URL:<http://www.exomi.com/products/smsgw.html>>
- /20/ **Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T.** *Hypertext Transfer Protocol – HTTP/1.1*. Request For Comments 2616, 1999.
- /21/ **Fowler, M., Scott, K.** *UML Distilled, Second Edition*. Addison-Wesley, 2000.
- /22/ **Gamma, E., Helm, R., Johnson, R., Vlissides, J.** *Design Patterns*. Addison-Wesley, 1995.
- /23/ **GNU.** *General Public License, Version 2*. 1991.
Saataavilla WWW-muodossa: <URL:<http://www.gnu.org/licenses/gpl.html>>
- /24/ **Hapner, M., Burridge, R., Sharma, R., Fialli, J., Stout, K.** *Java Message Service, Version 1.1*. Sun Microsystems, 2002.
- /25/ **IBM.** *WebSphere Application Server*. n.d.. (online) [viitattu 1.12.2004].
Saataavilla WWW-muodossa: <URL:<http://www-306.ibm.com/software/webservers/appserv/was/>>
- /26/ **ICSE.** *International Workshop on Distributed Event-Based Systems*. 26th International Conference on Software Engineering, 2004 (online) [viitattu 1.12.2004]. Saataavilla WWW-muodossa:
<URL:<http://serl.cs.colorado.edu/~carzanig/debs04/>>
- /27/ **ISO 8879.** *Standard Generalized Markup Language (SGML)*. International Organization for Standardization, 1986.
- /28/ **ISO/IEC 9075.** *Call-Level Interface (SQL/CLI)*. International Organization for Standardization, 1995.
- /29/ **ISO/IEC 9075.** *Database Language SQL with Integrity Enhancement*. International Organization for Standardization, 1989.
- /30/ **ISO/IEC 9075.** *Database Language SQL*. International Organization for Standardization, 1992.
- /31/ **ISO/IEC 9075.** *Database Language SQL*. International Organization for Standardization, 1999.
- /32/ **NCSA.** *The Common Gateway Interface Specification*. National Center for Supercomputing Applications Software Development Group.
Saataavilla WWW-muodossa:
<URL:<http://hoohoo.ncsa.uiuc.edu/cgi/interface.html>>
- /33/ **Microsoft.** *Internet Information Services*. 2004 (online) [viitattu 1.12.2004].
Saataavilla WWW-muodossa: <URL:<http://www.microsoft.com/iis>>
- /34/ **Microsoft.** *Microsoft Open Database Connectivity (ODBC)*. 2004 (online) [viitattu 1.12.2004].
Saataavilla WWW-muodossa: <URL:<http://msdn.microsoft.com/library/en-us/odbc/htm/dasdkodbcoverview.asp>>
- /35/ **Microsoft.** *SQL Server Architecture*. 2004 (online) [viitattu 1.12.2004].
Saataavilla WWW-muodossa:
<URL:<http://msdn.microsoft.com/SQL/sqlarchitecture/>>

- /36/ **Myers, J., Rose, M.** *Post Office Protocol – Version 3*. Request For Comments 1939, 1996.
- /37/ **MySQL.** *MySQL Database server*. 2004 (online) [viitattu 1.12.2004].
Saataavilla WWW-muodossa: <URL:http://www.mysql.com/products/mysql/>
- /38/ **MySQL.** *MySQL Reference Manual*. 2004 (online) [viitattu 1.12.2004].
Saataavilla WWW-muodossa: <URL:http://dev.mysql.com/doc/mysql/en/>
- /39/ **NIST.** *Integration Definition for Information Modeling (IDEFIX)*. National Institute of Standards and Technology, 1993.
- /40/ **Nokia.** *Nokia M2M Application Development Kit*. Datasheet. Nokia, 2002.
- /41/ **Nokia.** *Nokia 12 GSM module*. Datasheet. Nokia, 2004.
- /42/ **Nokia.** *Nokia 12i GSM module*. Datasheet. Nokia, 2004.
- /43/ **Nokia.** *Nokia 30 GSM Connectivity Terminal*. Datasheet. Nokia, 2004.
- /44/ **Nokia.** *Nokia 31 GSM Connectivity Terminal*. Datasheet. Nokia, 2003.
- /45/ **OMG.** *Common Object Request Broker Architecture (CORBA), v3.0.3*. Object Management Group, 2004.
- /46/ **Oracle.** *Oracle database*. 2004 (online) [viitattu 1.12.2004].
Saataavilla WWW-muodossa: <URL:http://www.oracle.com/database/>
- /47/ **Oracle.** *Oracle Application Server*. 2004 (online) [viitattu 1.12.2004].
Saataavilla WWW-muodossa: <URL:http://www.oracle.com/appserver/>
- /48/ **PHP.** *Hypertext Preprocessor*. n.d. (online) [viitattu 1.12.2004].
Saataavilla WWW-muodossa: <URL:http://www.php.net/>
- /49/ **Rescorla, E.** *HTTP Over TLS*. Request For Comments 2818, 2000
- /50/ **Rivest, R.** *The MD5 Message-Digest Algorithm*. Request For Comments 1321, 1992.
- /51/ **Schiller J.** *Mobiilitietoliikenne*. IT Press/Addison-Wesley, 2001.
- /52/ **Salin, M.** *Wireless remote monitoring system for a gaming machine*.
Diplomityö, Teknillinen korkeakoulu, Sähkö- ja tietoliikennetekniikan osasto, 2002.
- /53/ **Salin, M.** *Ztango M2M OTA Interface Specification*. Ztango, 2001.
- /54/ **Sandonato, R.** *Hardware Design Specification for Wireless Interface Module*.
Real Time & Embedded Systems Pty Ltd, 2000.
- /55/ **Siemens.** *AC45 Release 2.0 with SIM Access Profile*. Datasheet. Siemens, 2003.
- /56/ **Siemens.** *MC35i Terminal pre-release*. Datasheet. Siemens, 2003.
- /57/ **Siemens.** *MC55/MC56*. Datasheet. Siemens, 2003.
- /58/ **Siemens.** *TC35i Terminal pre-release*. Datasheet. Siemens, 2003.
- /59/ **Siemens.** *TC45 with GPRS & Java™*. Datasheet. Siemens, 2003.
- /60/ **Siemens.** *XT55 with GSM/GPRS & GPS preliminary*. Datasheet. Siemens, 2004.

- /61/ **Srinivasan, R.** *RPC: Remote Procedure Call Protocol Specification Version 2.* Request for Comments 1831, 1985.
- /62/ **Sun.** *Java Naming and Directory Interface™, Application Programming Interface (JNDI API).* Sun Microsystems, 1999.
- /63/ **Sun.** *Java Naming and Directory Interface™, Service Provider Interface (JNDI SPI).* Sun Microsystems, 1999.
- /64/ **Sun.** *Sun Java System Application Server.* Sun Microsystems, 2004 (online) [viitattu 1.12.2004]. Saatavilla WWW-muodossa:
<URL:http://www.sun.com/software/products/appsrvr/home_appsrvr.html>
- /65/ **Sun.** *Sun Java System Web Server.* Sun Microsystems, 2004 (online) [viitattu 1.12.2004]. Saatavilla WWW-muodossa:
<URL:http://www.sun.com/software/products/web_srvr/home_web_srvr.html>
- /66/ **Yeong, W., Howes, R., Kille, S.** *X.500 Lightweight Directory Access Protocol.* Request For Comments 1487, 1993.
- /67/ **Yeong, W., Howes, R., Kille, S.** *Lightweight Directory Access Protocol.* Request For Comments 1777, 1995.
- /68/ **Veitch, H.** *Functional Specification for Wireless Interface Module.* Real Time & Embedded Systems Pty Ltd, 2000.
- /69/ **WAP Forum.** *Wireless Application Protocol, Wireless Markup Language Specification, Version 1.3.* Wireless Application Protocol Forum, 2000.
- /70/ **Widom, J., Ceri S.** *Active Database Systems: Triggers and Rules For Advanced Database Processing.* Morgan Kaufman Publishers, 1996.
- /71/ **W3C.** *Cascading Style Sheets, level 1.* World Wide Web Consortium, 1999 (online) [viitattu 1.12.2004].
Saatavilla WWW-muodossa: <URL:<http://www.w3.org/TR/CSS1>>
- /72/ **W3C.** *Cascading Style Sheets, level 2, CSS2 Specification.* World Wide Web Consortium, 1998 (online) [viitattu 1.12.2004].
Saatavilla WWW-muodossa: <URL:<http://www.w3.org/TR/CSS2>>
- /73/ **W3C.** *HTML 4.01 Specification.* World Wide Web Consortium, 1999 (online) [viitattu 1.12.2004].
Saatavilla WWW-muodossa: <URL:<http://www.w3.org/TR/REC-html40/>>
- /74/ **W3C.** *XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition), A Reformulation of HTML 4 in XML 1.0.* World Wide Web Consortium, 2002 (online) [viitattu 1.12.2004].
Saatavilla WWW-muodossa: <URL:<http://www.w3.org/TR/xhtml1/>>
- /75/ **W3C.** *XHTML™ Basic.* World Wide Web Consortium, 2000 (online) [viitattu 1.12.2004].
Saatavilla WWW-muodossa: <URL:<http://www.w3.org/TR/xhtml-basic/>>
- /76/ **X/Open.** *Data Management: SQL Call Level Interface (CLI), C451.* X/Open Company, 1995.